НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Навчально-науковий інститут телекомунікаційних систем

Кафедра електронних комунікацій та інтернету речей

«На правах рукопису»
УДК _623.62:629.735:004.7

До захисту допущено:

Завідувач кафедри

_____ Анатолій МАКАРЕНКО

«___»_____20__ р.

Магістерська дисертація

на здобуття ступеня магістра

за освітньо-професійною програмою «Системи електронних комунікацій та Інтернету речей»

зі спеціальності 172 «Електронні комунікації та радіотехніка»

на тему: «Дослідження технологій радіоелектронної боротьби з БПЛА мереж ІОТ»

Виконав (-ла):
студент (-ка) ІІ курсу, групи ЦС-41мп
Галич Євген Олександрович _____

Науковий керівник:
к.т.н., доцент, доцент кафедри ЕКІР
Осипчук Сергій Олександрович _____

Рецензент:
к.т.н., с.н.с., доцент кафедри ТК
Міночкін Дмитро Анатолійович _____

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.
Студент (-ка) _____

Київ – 2025 року

Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Навчально-науковий інститут телекомунікаційних систем

Кафедра електронних комунікацій та інтернету речей

Рівень вищої освіти – другий (магістерський)

Спеціальність – 172 «Електронні комунікації та радіотехніка»

Освітньо-професійна програма «Системи електронних комунікацій та Інтернету речей»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Анатолій МАКАРЕНКО

«___»_____20__ р.

ЗАВДАННЯ
на магістерську дисертацію студенту

Галич Євген Олександрович

1. Тема дисертації «Дослідження технологій радіоелектронної боротьби з БПЛА мереж ІОТ», науковий керівник дисертації Осипчук Сергій Олександрович, кандидат наук, доцент, затверджені наказом по університету від «03» листопада 2025 р. №4777-с

2. Термін подання студентом дисертації: 08 грудня 2025

3. Об’єкт дослідження є процеси виявлення, класифікації та нейтралізації сигналів БПЛА та ІОТ-пристроїв у середовищі радіоелектронної боротьби.

4. Предмет дослідження інструменти та методи РЕБ, що дозволяють виявляти, класифікувати та нейтралізувати сигнали БПЛА та ІОТ-пристроїв

5. Перелік завдань, які потрібно розробити:

1. Розробка концептуальної архітектури системи РЕБ для ІОТ-середовища.
2. Розробка ефективних методів виявлення та класифікації сигналів БПЛА в умовах ІОТ.
3. Проектування модулю прийняття рішень на основі аналізу загроз.
4. Моделювання та тестування системи РЕБ у змодельованих умовах ІОТ.

6. Перелік графічного (ілюстративного) матеріалу: діаграма діяльності, діаграма послідовності, діаграма компонентів, контекстна діаграма та рисунки інтерфейсу та результатів роботи програми.

7. Орієнтовний перелік публікацій: Austin, Reg (2010). Unmanned Aircraft Systems: UAVs Design, Development, and Deployment. Chichester: Wiley. ISBN 978-0-470-66480-3. OCLC 656287984., Zong, J., Zhu, B., Hou, Z., Yang, X., & Zhai, J. (2021). Evaluation and Comparison of Hybrid Wing VTOL UAV with Four Different Electric Propulsion Systems. Aerospace, 8(9), 256., EW 101: A First Course in Electronic Warfare; David Adamy (2001)., та інші онлайн - статті.

8. Дата видачі завдання 16.11.2024

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1	Вибір та формулювання теми дисертації	16.11.2024	виконано
2	Огляд літератури та аналіз існуючих досліджень у відповідній темі	01.09.2025	виконано
3	Написання теоретичної частини дослідження	07.10.2025	виконано
4	Створення та тестування програмного забезпечення	17.11.2025	виконано
5	Проведення експериментальних досліджень і тестування	23.11.2025	виконано
6	Написання результатів досліджень та аналіз отриманих даних	29.11.2025	виконано
7	Оформлення магістерської дисертації	08.12.2025	виконано

Студент Євген ГАЛИЧ

Науковий керівник Сергій ОСИПЧУК

РЕФЕРАТ

Дисертація складається з 107 сторінки, містить 30 ілюстрацій, 8 таблиць, 3 додатки та 29 джерел у переліку посилань.

Тема дослідження: “Дослідження технологій радіоелектронної боротьби з БПЛА мереж ІОТ”.

З ростом кількості безпілотних літальних апаратів та ІОТ-пристроїв зростає також загроза їх використання для несанкціонованих або ворожих операцій. Технології РЕБ, зокрема для боротьби з БПЛА в ІОТ-середовищі, набувають особливої важливості. Проблема створення ефективних систем РЕБ для таких умов потребує подальших досліджень та розробок, що обґрунтовує актуальність цієї роботи.

Метою роботи є розробка ефективної системи РЕБ для протидії БПЛА в умовах ІОТ, що передбачає використання новітніх методів виявлення, класифікації та нейтралізації загроз. Для досягнення цієї мети було вирішено кілька задач: розробка математичних моделей сигналів БПЛА та ІОТ-пристроїв, створення програмного забезпечення для емуляції технологій РЕБ, а також тестування та оцінка ефективності розробленої системи.

Об’єктом дослідження є процеси виявлення, класифікації та нейтралізації сигналів БПЛА та ІОТ-пристроїв у середовищі радіоелектронної боротьби.

Предметом дослідження є інструменти та методи РЕБ, що дозволяють виявляти, класифікувати та нейтралізувати сигнали БПЛА та ІОТ-пристроїв.

Методи дослідження: спектральний аналіз, математичне моделювання, машинне навчання (підтримувальні векторні машини для класифікації сигналів), а також моделювання процесів РЕБ.

Науковою новизною є розробка нових математичних моделей для генерації та класифікації сигналів БПЛА та ІОТ-пристроїв, а також

створення модульної архітектури програмного забезпечення для радіоелектронної боротьби в умовах IOT.

Практичне значення одержаних результатів: розроблена система може бути впроваджена в охоронні системи для забезпечення захисту від безпілотних літальних апаратів та інших пристроїв у середовищі IOT.

Ключові слова: радіоелектронна боротьба, безпілотні літальні апарати, Інтернет речей, класифікація сигналів, контрзаходи, спектральний аналіз, машинне навчання, програмне забезпечення, захист від радіоелектронних атак.

ABSTRACT

The dissertation consists of 107 pages, contains 30 illustrations, 8 tables, 3 appendix and 29 sources in the list of references.

Research topic: “Research on technologies of electronic warfare against UAVs of IOT networks”.

With the growth of the number of unmanned aerial vehicles and IOT devices, the threat of their use for unauthorized or hostile operations also increases. EW technologies, in particular for combating UAVs in the IOT environment, are gaining particular importance. The problem of creating effective EW systems for such conditions requires further research and development, which justifies the relevance of this work.

The purpose of the work is to develop an effective EW system to counter UAVs in IOT conditions, which involves the use of the latest methods of detecting, classifying and neutralizing threats. To achieve this goal, several tasks were solved: the development of mathematical models of UAV and IOT device signals, the creation of software for emulating EW technologies, as well as testing and evaluating the effectiveness of the developed system.

The object of the research is the processes of detection, classification and neutralization of UAV and IOT device signals in the electronic warfare environment.

The subject of the research is EW tools and methods that allow detecting, classifying and neutralizing UAV and IOT device signals.

Research methods: spectral analysis, mathematical modeling, machine learning (support vector machines for signal classification), as well as modeling of EW processes.

The scientific novelty is the development of new mathematical models for generating and classifying UAV and IOT device signals, as well as the creation of a modular software architecture for electronic warfare in IOT conditions.

Practical significance of the results obtained: the developed system can be implemented in security systems to provide protection against unmanned aerial vehicles and other devices in the IOT environment.

Keywords: electronic warfare, unmanned aerial vehicles, Internet of Things, signal classification, countermeasures, spectral analysis, machine learning, software, protection against electronic attacks.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ І ТЕРМІНІВ

БПЛА - безпілотний літальний апарат

IOT - Інтернет речей

РЕБ - радіоелектронна боротьба

PEI - радіоелектронний вплив

DEW - Системи направленої енергії (Directed Energy Weapons)

GPS - глобальна система позиціонування (Global Positioning System)

SNR - співвідношення сигнал/шум

J/S - співвідношення джемінг/шум

SVC - підтримувальна векторна машина (Support Vector Classification)

SINR - співвідношення сигнал/інтерференція+шум

FFT - швидке перетворення Фур'є (Fast Fourier Transform)

ВСТУП

Сучасний розвиток БПЛА та ІОТ значно змінив сферу технологій і комунікацій, відкривши нові можливості для їх застосування у різноманітних галузях, від військових операцій до цивільного використання. Зокрема, БПЛА мають величезний потенціал у задачах спостереження, моніторингу, доставки та навіть в умовах боротьби з природними катастрофами. Разом із цим зростає і загроза їх використання для несанкціонованих або ворожих операцій, що ставить під загрозу безпеку державних, військових та комерційних об’єктів. Актуальність дослідження визначається необхідністю створення нових методів і технологій для протидії загрозам, що виникають через широке поширення БПЛА та пристроїв ІОТ. Зокрема, сучасні системи РЕБ повинні бути адаптовані до умов, де одночасно працюють різні типи сигналів, таких як сигнали від БПЛА і ІОТ-пристроїв, що можуть перехресно впливати на ефективність системи захисту. У зв’язку з цим важливим є розроблення системи РЕБ, яка б могла ефективно виявляти, класифікувати та нейтралізувати загрози в таких умовах.

Метою роботи є розробка ефективної системи РЕБ для протидії БПЛА в умовах ІОТ, що передбачає використання новітніх методів виявлення, класифікації та нейтралізації загроз. Для досягнення цієї мети було вирішено кілька задач: розробка математичних моделей сигналів БПЛА та ІОТ-пристроїв, створення програмного забезпечення для емуляції технологій РЕБ, а також тестування та оцінка ефективності розробленої системи.

Об’єктом дослідження є процеси виявлення, класифікації та нейтралізації сигналів БПЛА та ІОТ-пристроїв у середовищі радіоелектронної боротьби.

Предметом дослідження є інструменти та методи РЕБ, що дозволяють виявляти, класифікувати та нейтралізувати сигнали БПЛА та ІОТ-пристроїв.

Методи дослідження: спектральний аналіз, математичне моделювання, машинне навчання (підтримувальні векторні машини для класифікації сигналів), а також моделювання процесів РЕБ.

Науковою новизною є розробка нових математичних моделей для генерації та класифікації сигналів БПЛА та ІОТ-пристроїв, а також створення модульної архітектури програмного забезпечення для радіоелектронної боротьби в умовах ІОТ.

Практичне значення одержаних результатів: розроблена система може бути впроваджена в охоронні системи для забезпечення захисту від безпілотних літальних апаратів та інших пристроїв у середовищі ІОТ.

1 АНАЛІЗ ТЕХНОЛОГІЙ РАДІОЕЛЕКТРОННОЇ БОРОТЬБИ ТА ОСОБЛИВОСТЕЙ ІОТ

1.1 Огляд сучасних безпілотних літальних апаратів та принципів їх зв’язку

Безпілотні літальні апарати (БПЛА) - це технічні пристрої, які здатні виконувати польоти в повітряному просторі без наявності пілота на борту[1]. Завдяки розвитку технологій вони стали важливими інструментами в різних сферах, включаючи військову, комерційну, наукову діяльність та моніторинг навколишнього середовища. У широкому спектрі застосувань БПЛА використовуються як для збору даних, так і для виконання конкретних завдань, таких як доставлення вантажів, спостереження, а також участь у складних бойових операціях[1].

Інтеграція БПЛА в сучасні технологічні системи значно змінила багато аспектів взаємодії техніки та людини. Безпілотні літальні апарати мають низку характеристик, які визначають їх здатність виконувати різноманітні завдання. Сучасні БПЛА зазвичай характеризуються високою маневреністю, можливістю тривалого перебування в повітрі, здатністю здійснювати точні вимірювання та збір даних, а також можливістю бути ефективними в умовах складних навколишніх середовищ[1]. На рис.1.1 відображено БПЛА типу VTOL.

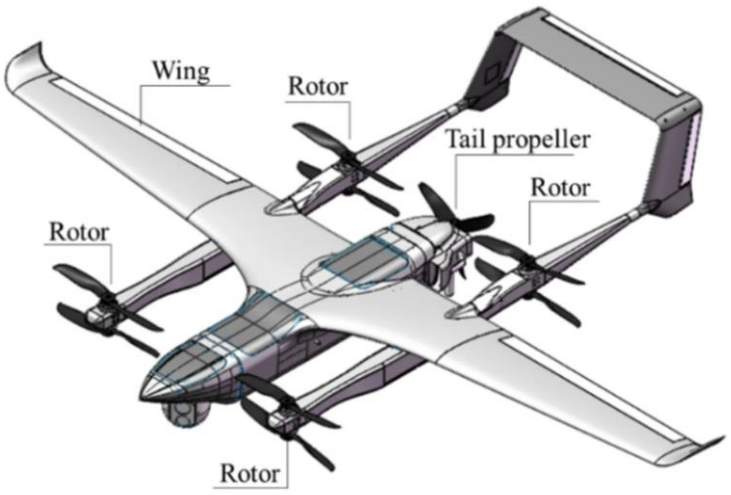


Рисунок 1.1 - БПЛА типу VTOL[2]

Безпілотні літальні апарати можна класифікувати за кількома критеріями. Одним з основних критеріїв є тип конструкції: апарати з фіксованим крилом, мультикоптери та гібридні апарати, які можуть злітати й приземлятися вертикально (VTOL - вертикальний зліт і посадка)[3]. На рис.1.2 відображено БПЛА типу мультикоптер.

Рисунок 1.2 - БПЛА типу мультикоптер [4]

Мультикоптер - це безпілотний літальний апарат, який використовує декілька роторів для підйому і маневрування в повітрі[4]. Мультикоптери відрізняються від традиційних літальних апаратів тим, що мають кілька гвинтів (ротори), які забезпечують підйом і стабільність. Завдяки цьому вони можуть бути дуже маневровими та зручними для виконання різноманітних завдань, таких як спостереження, аерофотозйомка, доставка вантажів та інше.

Фіксоване крило - це тип безпілотного літального апарата, який має фіксовані крила, що дозволяють йому генерувати підйомну силу за

рахунок аеродинамічного тиску на поверхню крил під час руху через атмосферу[4]. На рис.1.3 відображено БПЛА типу фіксоване крило.

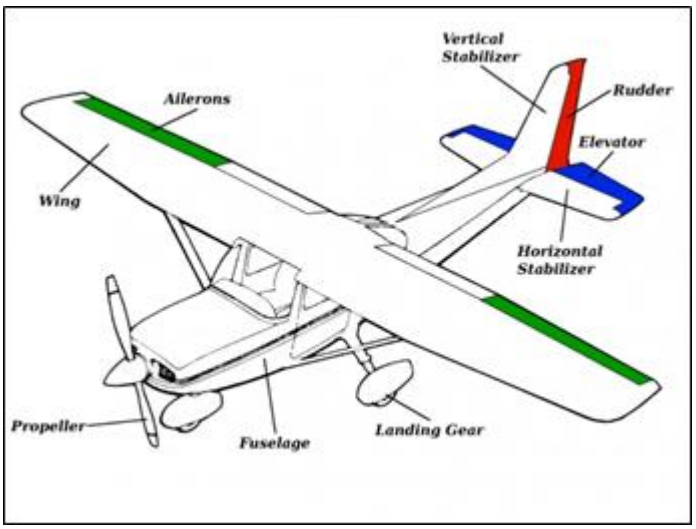


Рисунок 1.3 - Фіксоване крило бпла [4]

Відмінною рисою такого БПЛА є те, що він не використовує пропелери для вертикального зльоту і посадки, як мультикоптери, а замість цього зазвичай потребує розбігу для зльоту і пробігу для приземлення. У табл.1.1 відображено класифікацію типів БПЛА.

Таблиця 1.1 - Класифікація типів БПЛА

Критерій	Тип БПЛА	Опис
Тип конструкції	Фіксоване крило	Апарат має стабільні крила, схожі на звичайні літаки.
	Мультикоптер	БПЛА з кількома роторами (від 4 і більше).
	Гібрид (VTOL)	Комбіновані апарати, здатні здійснювати вертикальний зліт і посадку, мають фіксоване крило та вертикальні двигуни.

* сформовано на основі [5]

Продовження таблиці 1.1

Критерій	Тип БПЛА	Опис
Вага	Легкі (до 5 кг)	Малі БПЛА, часто використовуються для цивільних та спортивних потреб.
	Середні (5-25 кг)	Підходять для більш складних завдань, таких як аерофотозйомка, розвідка та моніторинг.
	Важкі (понад 25 кг)	Використовуються для професійних завдань, наприклад, доставки вантажів, військових операцій.
Вантажопідйомність	Легкі (до 2 кг)	Малі БПЛА, які зазвичай використовуються для фільмування, моніторингу навколишнього середовища.
	Середні (2-10 кг)	БПЛА, що здатні нести важчі навантаження (наприклад, камери високої роздільної здатності).
	Важкі (понад 10 кг)	Професійні БПЛА для важких завдань, таких як доставка вантажів або використання в військових операціях.
Призначення	Цивільні	Використовуються для картографії, моніторингу довкілля, агрономії, аерофотозйомки.

* сформовано на основі [5]

Зв'язок є одним з основних аспектів, що визначає ефективність функціонування безпілотних літальних апаратів. Системи зв'язку БПЛА включають різні технології, що дозволяють обмінюватися даними між апаратом і операторами. Як правило, цей зв'язок здійснюється через радіочастоти, але для великих відстаней можуть використовуватися супутникові канали зв'язку або лінії видимості. Канали зв'язку можуть бути розділені на кілька категорій: канал керування, канал передачі телеметрії та канал передачі даних.

Канал керування забезпечує безперервне управління літальним апаратом[6], що включає в себе команди для зльоту, посадки, маневрів і

корекції курсу. Зв’язок у цьому каналі вимагає високої надійності та низьких затримок, оскільки навіть незначні проблеми в комунікації можуть призвести до аварії або втрати контролю над апаратом.

Канал передачі даних відповідає за забезпечення зворотного зв'язку з оператором включає передавання відео з камер, отриманих даних із сенсорів (тепловізори, гідролокація), а також телеметрії про стан апарата. Висока пропускна здатність і надійність цього каналу критичні для виконання більшості розвідувальних та бойових завдань[6].

Канал телеметрії збирав дані про стан систем БПЛА, такі як швидкість, висота, температура двигунів, рівень заряду батареї. Інформація про технічний стан апарата є важливою для коректного управління, а також для прогнозування можливих поломок або необхідності ремонту. Для забезпечення надійної комунікації використовуються різноманітні методи захисту сигналу, а також технології, спрямовані на підвищення стійкості до перешкод, такі як частотне розмноження, використання широкосмугових каналів зв’язку та змішання сигналів[6].

Зв'язок БПЛА з мережею Інтернету речей є важливим напрямом розвитку цих систем. IOT дозволяє інтегрувати БПЛА в складні мережі, де дрони можуть працювати в якості мобільних платформ для збору та передачі даних із сенсорів, камер та інших пристроїв. У таких мережах БПЛА можуть виступати в ролі тимчасових або постійних «вузлів» для збору даних із різних точок і їх передавання до центральної системи або в інші пристрої мережі[6]. Важливою перевагою цієї інтеграції є мобільність БПЛА, що дає можливість охоплювати великі території або важкодоступні місцевості, де традиційна інфраструктура IOT не може бути розгорнута. Наприклад, БПЛА можуть виступати в ролі переносних базових станцій для мобільних сенсорів або як вузли зв'язку в ситуаціях, де інші технології можуть бути неефективними. Удосконалення таких мереж потребує високої пропускної здатності каналу зв'язку для забезпечення стабільного

обміну даними, а також адаптивності до змінюваних умов, наприклад, до змін у навколишньому середовищі або в топології мережі[6].

В умовах сучасних військових та безпекових загроз одним із ключових напрямів є розробка методів захисту БПЛА від радіоелектронної боротьби. РЕБ (рис. 1.4) включає в себе комплекс заходів, що передбачають глушіння, перехоплення або навіть маніпулювання сигналами, які використовуються для управління та передачі даних від БПЛА[7]. Це ставить перед розробниками БПЛА вимоги до підвищення стійкості та безпеки зв'язку, застосування криптографічних протоколів та застосування технологій, які дозволяють дрону залишатись у безпеці навіть за наявності активної радіоелектронної боротьби.

Рисунок 1.4 - РЕБ [8]

Система захисту від РЕБ повинна бути інтегрована у загальну архітектуру БПЛА, при цьому особливу увагу слід звернути на створення резервних каналів зв'язку та засобів подолання перешкод. Безпілотні літальні апарати є одними з найперспективніших технологічних інструментів сучасності, що знайшли широке застосування в різних сферах

діяльності[8]. Принципи їхнього зв'язку, від ефективності яких значною мірою залежить успішність виконання завдань, постійно розвиваються[8]. Зокрема, інтеграція БПЛА в мережі Інтернету речей відкриває нові можливості для створення складних, адаптивних та мобільних комунікаційних систем. Водночас, не менш важливим є захист зв'язку від радіоелектронної боротьби, оскільки можливість перехоплення або заглушення сигналів може призвести до значних втрат.

1.2 Особливості функціонування мереж IOT та їх вразливості до радіоелектронного впливу

IOT - це система взаємозв'язаних фізичних об'єктів, пристроїв і сенсорів, здатних збирати і обмінювати даними через мережі без безпосередньої участі людини[9]. IOT дозволяє автоматизувати процеси та знижувати потребу в ручному управлінні, створюючи систему інтелектуальних зв'язків між об'єктами, які здатні самостійно виконувати операції на основі зібраних даних. Основною метою IOT є оптимізація процесів, поліпшення продуктивності, збільшення ефективності та забезпечення більш точного контролю за фізичними об'єктами.

Принцип роботи таких мереж полягає в тому, що фізичні об'єкти, які складають IOT, оснащуються різноманітними сенсорами, пристроями для передачі та отримання даних, а також мікропроцесорами для обробки отриманої інформації[9]. Дані з сенсорів, які вимірюють фізичні величини (температуру, вологість, тиск, місце знаходження тощо), передаються через мережу до централізованого сервера або в інші пристрої для подальшої обробки. Інтернет речей (рис.1.5) охоплює велику кількість різноманітних пристроїв, які можуть бути застосовані в різних сферах: від побутових пристроїв, таких як розумні термостати і домашні системи безпеки, до промислових застосувань - таких як системи управління

виробничими процесами, розумні міста, медичні пристрої для моніторингу здоров'я та багато інших[9].

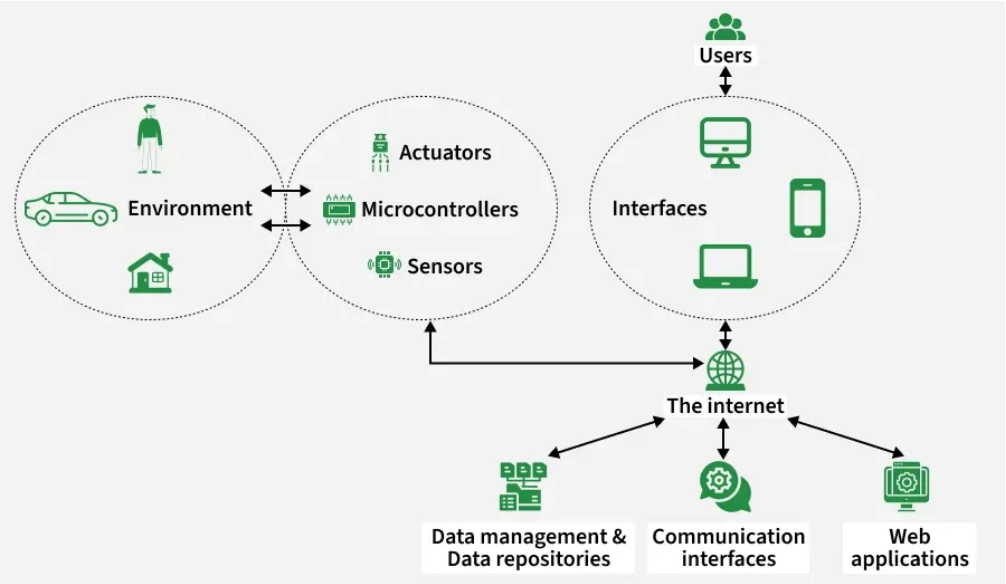


Рисунок 1.5 - Інтернет речей [10]

Технологічні рішення IOT охоплюють численні сфери, завдяки чому їх попит на сьогодні є одним із найвищих серед новітніх технологій. Система IOT включає три основні компоненти: пристрої (сенсори, датчики), мережу для обміну даними та програмне забезпечення для обробки отриманої інформації[10]. Мережа IOT базується на різних типах комунікаційних протоколів, таких як Wi-Fi, Bluetooth, Zigbee, LoRaWAN, 5G тощо, що дозволяють передавати дані між пристроями, забезпечуючи безперебійний зв'язок і інтеграцію між фізичними об'єктами та цифровими системами[10].

Пристрої IOT оснащуються сенсорами, що забезпечують збір даних про навколишнє середовище або про стан об'єкта, до якого вони прикріплені. Це можуть бути датчики руху, температури, вологості, тиску, відеокамери або спеціалізовані пристрої для вимірювання певних фізичних характеристик[10]. Сенсори передають зібрані дані через мережу до центрального процесора або серверу для подальшого аналізу. Програмне забезпечення в свою чергу може приймати рішення на основі цих даних,

автоматизуючи процеси і забезпечуючи функціонування системи без участі людини.

Одним із ключових аспектів функціонування IOT є зв'язок між пристроями. Передача даних між ними здійснюється через стандартні мережеві протоколи, що забезпечують швидкість, безпеку та ефективність комунікації. Кожен пристрій може передавати дані іншим пристроям або до центрального сервера для подальшої обробки, що дозволяє здійснювати зворотний зв'язок і автоматично коригувати роботу пристроїв. Інтернет речей стикається з численними викликами в контексті безпеки[10]. Незважаючи на безперервний розвиток технологій захисту даних, мережі IOT залишаються вразливими до різноманітних кіберзагроз. Однією з основних проблем є недосконаліть захисту на рівні пристроїв. Багато IOT-пристроїв, особливо тих, що мають низьку вартість і просту конструкцію, не забезпечують належного рівня захисту інформації, що передається або зберігається на них. Це створює можливість для несанкціонованого доступу до конфіденційних даних. До найбільш поширених загроз можна віднести[10]:

- неавторизований доступ до пристроїв або їх злом. Багато пристроїв IOT не мають надійної аутентифікації, що робить їх уразливими до атак;
- інтерцепція даних під час передачі через мережу, що може призвести до витоку важливої інформації;
- масштабні атаки на інфраструктуру. Пристрій, що є частиною IOT, може бути використаний для атак на інші компоненти мережі або для виконання дистрибуції вірусів.

Однією з найбільших проблем для безпеки IOT є вразливість до радіоелектронного впливу (PEI). Це явище виникає, коли електронні пристрої або мережі зазнають втручання з боку сторонніх радіочастотних сигналів, що може викликати різні негативні ефекти, від порушення зв'язку

до повної несанкціонованої зміни даних, що передаються. PEI включає в себе низку різних впливів, таких як[11]:

1. Глушіння сигналу - намагання перервати комунікаційний канал між пристроями IOT.
2. Перехоплення сигналу - за допомогою спеціалізованих засобів можна отримувати радіосигнали, що передаються між пристроями, і зловмисно використовувати ці дані.
3. Модифікація сигналу - несанкціоноване втручання в інформацію, що передається, для змінення її змісту або команди.
4. Радіоелектронне розвідка - використання специфічних приймачів для збору інформації про мережу IOT без її відомості.

Існують різні підходи до забезпечення захисту мереж IOT від радіоелектронного впливу. Одним із них є шифрування даних, що передаються між пристроями. Використання криптографічних методів шифрування дозволяє забезпечити конфіденційність даних, навіть у разі перехоплення сигналу. Також важливим аспектом є використання спеціалізованих протоколів зв'язку, які мають високу стійкість до глушіння або перехоплення сигналів. Наприклад, для безпечних IOT-систем можуть використовуватися протоколи з вбудованими засобами аутентифікації, що дозволяють визначити ідентичність користувачів або пристроїв.

Варто також зазначити важливість моніторингу спектра радіочастот, що дозволяє виявляти аномалії або несанкціоноване використання радіочастотних каналів. Для захисту від атак в реальному часі використовуються системи активного захисту, що дозволяють автоматично виявляти і блокувати шкідливі сигнали. Мережі IOT мають величезний потенціал у різних сферах діяльності, однак для забезпечення їх стабільної та безпечної роботи необхідно удосконалювати механізми захисту. Враховуючи високий рівень вразливості до радіоелектронного впливу, важливо продовжувати розвиток нових методів захисту і підвищувати стійкість мереж IOT до атак.

1.3 Існуючі методи радіоелектронної боротьби з БПЛА та постановка задачі дослідження

Розвиток безпілотних літальних апаратів став важливим етапом у технологічному прогресі. Безпілотні апарати використовуються в різноманітних сферах - від цивільних завдань, таких як доставка вантажів і моніторинг навколишнього середовища, до військових операцій, розвідки, коригування вогню і навіть атаки на ворога. Однак збільшення використання БПЛА також викликає значні загрози для безпеки, оскільки ці апарати можуть бути використані в злочинних або терористичних цілях. Внаслідок цього виникає потреба в розробці та впровадженні ефективних засобів радіоелектронної боротьби , здатних нейтралізувати чи знищити ці апарати. Радіоелектронна боротьба з БПЛА включає низку методів, спрямованих на порушення їхнього функціонування шляхом впливу на системи зв'язку, навігації та управління. У цьому контексті важливо розглядати існуючі методи РЕБ, а також визначити проблеми та перспективи їх удосконалення для забезпечення високої ефективності захисту від безпілотних літальних апаратів.

Методи РЕБ, що використовуються для боротьби з безпілотними літальними апаратами, можна умовно поділити на кілька основних напрямів: глушіння радіозв'язку, перехоплення сигналів управління, застосування спеціалізованих засобів для нейтралізації БПЛА, а також вплив на навігаційні системи. У даному розділі розглянемо основні існуючі методи РЕБ, які використовуються для боротьби з БПЛА, а також сформулюємо задачі для подальших досліджень.

1.3.1 Глушіння радіозв'язку

Глушіння сигналу є одним із найпоширеніших методів радіоелектронної боротьби з БПЛА. Це метод, при якому спеціалізовані пристрої створюють радіоінтерференцію, що заважає нормальній роботі

БПЛА[12]. Залежно від типу та потужності пристроїв, глушіння може мати різні характеристики.

Широкосмугове глушіння (Wideband Jamming) - створюється потужний сигнал на великому спектрі частот, що може заглушити більшу кількість каналів одночасно. Такий метод ефективний для зупинки БПЛА, що використовують різні частотні діапазони для зв'язку та навігації[13].

Буковель-AD (рис.1.6) - українська система протидії БПЛА (C-UAV), розроблена компанією Proximus LLC. Її завдання - виявлення безпілотників і глушіння каналів зв'язку та навігації (GPS/GLONASS/Galileo/BeiDou) між дроном та оператором.

Рисунок 1.6 - Українська система Bukovel-AD [14]

За характеристиками: виявлення - до ~70–100 км (залежно від типу БПЛА), радіус придушення (джемінгу) - ~15–20 км[14]. Вузькосмугове глушіння (Narrowband Jamming) - це глушення сигналу з обмеженим діапазоном частот, зокрема на специфічних частотах, які використовуються конкретним БПЛА[13]. Цей метод дозволяє ефективно націлюватися на певні сигнали, зменшуючи вплив на інші радіосигнали.

Рисунок 1.7 - DroneGun Mk4 [15]

На рис.1.7 відображено систему «DroneGun» , що здійснює вузькосмугове глушіння на частотах, що використовуються в системах управління БПЛА, таких як Wi-Fi або частоти мобільних мереж[15]. «Пістолет»-джамер, що створює РЧ-інтерференцію - порушує зв’язок управління/канали передачі даних дронів.

1.3.2 Перехоплення управлінських сигналів

Іншим важливим методом РЕБ є перехоплення управлінських сигналів, які передаються між оператором і БПЛА. Завдяки цьому зловмисник може захопити контроль над БПЛА, підмінити команди або навіть перенаправити його рух в іншу точку. Цей метод потребує спеціалізованого обладнання, яке здатне перехоплювати радіосигнали на певних частотах і підміняти їх, таким чином змінюючи функціонування БПЛА. Перехоплення управлінських сигналів може бути здійснене через

використання технологій для дешифрування сигналу або за допомогою засобів для захоплення і повторної передачі команд управління. Це дає можливість зловмиснику або атакуючій стороні отримати доступ до важливих параметрів управління БПЛА і змусити його виконувати несанкціоновані маневри. У випадку з перехопленням, система «Skylock» (рис. 1.8) може перехоплювати частоти, на яких працюють БПЛА, і передавати відповідні сигнали для відключення зв'язку[16].

Рисунок 1.8 - Антидронова система «Skylock» [16]

Особливо небезпечним є цей метод у випадках, коли БПЛА використовують слабко захищені канали зв'язку або не мають вбудованих засобів аутентифікації для перевірки автентичності сигналів. Крім того, за допомогою перехоплення можна не лише змінювати управлінські команди, але й здійснювати більш складні атаки, такі як захоплення або навіть знищення БПЛА за допомогою спеціальних атак.

1.3.3 Спеціалізовані засоби для нейтралізації БПЛА

У сучасних умовах, коли звичайні методи глушіння або перехоплення можуть бути недостатньо ефективними, з'являється потреба в застосуванні спеціалізованих засобів для фізичної нейтралізації БПЛА. Це можуть бути системи, здатні не лише припинити зв'язок з БПЛА, а й знищити його або змусити приземлитися. Системи направленої енергії (Directed Energy Weapons, DEW) - вони використовують високочастотне або лазерне випромінювання для того, щоб вплинути на електроніку БПЛА, вивести її з ладу або навіть викликати вибух[17].

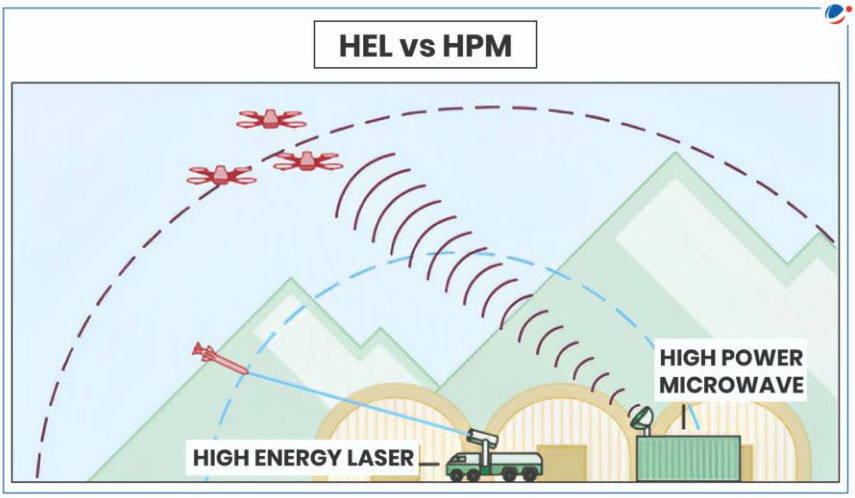


Рисунок 1.9 - Системи направленої енергії [17]

Деякі розробки передбачають використання мереж, які можуть викидатися з пристроїв для захоплення БПЛА в повітрі. Такі системи дозволяють фізично зупиняти дрони і сприймаються як ефективні в умовах міських середовищ або в зонах з високою концентрацією дронів. За допомогою спеціалізованих технологій можна порушити стабільність польоту БПЛА, змусити його здійснити аварійну посадку або повернути на точку запуску.

Безпілотні літальні апарати часто використовують супутникові навігаційні системи, такі як GPS, для визначення свого місця розташування

та коригування траєкторії польоту. Порушення або глушіння сигналу GPS є одним із найефективніших способів впливу на БПЛА. У сучасних умовах БПЛА, які працюють без постійного контролю оператора, можуть повністю втратити орієнтацію, якщо сигнал GPS буде заблоковано або спотворено.

Дії, спрямовані на глушіння GPS або імплантацію фальшивих навігаційних сигналів (Spoofing), можуть призвести до значних порушень у функціонуванні БПЛА, включаючи втрату стабільності польоту, несанкціоноване переміщення або навіть аварії. Для протидії цьому методу необхідно розробляти нові технології, такі як використання альтернативних систем навігації, інтегрованих з різноманітними датчиками, що дозволяють компенсувати втрату GPS.

1.3.4 Постановка задачі дослідження

У межах дослідження та розробки системи радіоелектронної боротьби для захисту від загроз, пов'язаних з безпілотними літальними апаратами у середовищах Інтернету речей, необхідно сформулювати кілька ключових завдань, що мають забезпечити ефективність і масштабованість цієї системи. Оскільки БПЛА все частіше використовуються в різних сферах - від цивільних застосувань до військових операцій - система РЕБ повинна бути здатною виявляти, класифікувати і нейтралізувати ці загрози в реальному часі з урахуванням обмежень, пов'язаних із специфікою IOT.

Основні задачі дослідження включають:

1. Розробка концептуальної архітектури системи РЕБ для IOT-середовища.
2. Розробка ефективних методів виявлення та класифікації сигналів БПЛА в умовах IOT.
3. Проектування модулю прийняття рішень на основі аналізу загроз.

4. Моделювання та тестування системи РЕБ у змодельованих умовах IOT.

1.4 Висновки до розділу 1

У першому розділі було здійснено аналіз сучасних безпілотних літальних апаратів та принципів їх зв'язку, а також розглянуті основні аспекти їх функціонування в умовах радіоелектронної боротьби . Окреслено основні методи РЕБ, які використовуються для протидії БПЛА, та їх ефективність у різних сценаріях.

Аналіз показав, що БПЛА є ключовими елементами сучасних технологій з великою кількістю застосувань як у цивільній, так і в військовій сферах. Вони здатні виконувати широкий спектр завдань завдяки своїм характеристикам: маневреності, високій автономії та можливості використання різних типів навігації та зв'язку. Проте зростаюча загроза використання БПЛА для проведення несанкціонованих або ворожих операцій вимагає розробки ефективних методів радіоелектронної боротьби.

Основні методи РЕБ включають глушіння радіозв'язку, перехоплення управлінських сигналів, а також фізичну нейтралізацію БПЛА через використання спеціалізованих засобів, таких як системи направленої енергії чи механічні засоби захоплення. Ці методи є ефективними, але мають певні обмеження, зокрема залежно від типу сигналу, дальності та потужності використовуваних систем.

Інтеграція БПЛА в середовище IOT створює нові можливості для застосування дронів як мобільних платформ для збору та передачі даних, що зумовлює додаткові вимоги до систем захисту від радіоелектронного впливу. У такому контексті необхідно враховувати різноманітність сигналів від БПЛА та IOT-пристроїв, а також виклики, пов'язані з перехресною інтерференцією між ними. Визначено, що для ефективної протидії БПЛА в умовах IOT необхідна розробка комплексної системи РЕБ, що поєднує різні методи виявлення, класифікації та нейтралізації

загроз, а також інтеграцію з іншими системами безпеки. Ця система повинна включати модульний підхід для забезпечення гнучкості та адаптивності до змінюваних умов та різноманітних загроз.

У першому розділі також було сформульовано постановку завдання дослідження, що полягає в розробці ефективних методів РЕБ для протидії БПЛА в умовах мереж ІОТ.

2 МОДЕЛЮВАННЯ ТА ПРОЄКТУВАННЯ СИСТЕМИ РЕБ В УМОВАХ IOT

2.1 Концептуальна архітектура системи радіоелектронної боротьби

Концептуальна архітектура системи радіоелектронної боротьби в умовах IOT побудована на модульному підході, що забезпечує ефективне виявлення, оцінку та реагування на потенційні загрози, пов'язані з використанням БПЛА в мережах IOT. Система складається з кількох основних компонентів:

1. Підсистема спостереження (Sensors) - здійснює збір даних з різних джерел, включаючи радіочастотні сенсори, мережеві монітори IOT, а також оптичні чи інфрачервоні сенсори. Ці сенсори забезпечують збирання інформації про навколишнє середовище та підозрілі об'єкти в реальному часі.
2. Підсистема виявлення та класифікації (Detection & Classification) - відповідає за обробку зібраних даних, виділення потенційно небезпечних об'єктів, їх класифікацію за типами (визначення типу БПЛА) та оцінку рівня ризику. Цей компонент застосовує алгоритми для точного виявлення загроз і їх швидкої класифікації.
3. Модуль злиття даних (Data Fusion) - інтегрує інформацію з різних сенсорів для створення єдиної повної картини ситуації. Це дозволяє системі враховувати різні типи даних, об'єднуючи їх в єдиний контекст для подальшої обробки.
4. Модуль прийняття рішень (Decision & Control) - на основі зібраних та оброблених даних визначає необхідні реакції системи відповідно до заданих політик безпеки. Цей модуль також включає інтерфейс для оператора, що дозволяє йому контролювати і коригувати процеси реагування в разі необхідності.
5. Модуль пом'якшення загроз (Mitigation) - забезпечує безпечне та законне реагування на виявлені загрози. Він включає в себе механізми інформування оператора про потенційні небезпеки та

ініціацію процедур для безпечного відведення або нейтралізації об'єктів, наприклад, відправлення команд для приземлення БПЛА або його знищення.

6. Підсистема управління та інтеграції (C2 / Integration) - відповідає за взаємодію з зовнішніми IOT-платформами та іншими інформаційними системами. Вона також забезпечує логування подій і аудит, що дозволяє здійснювати моніторинг системи та зберігати необхідну документацію для аналізу та звітності.

Ця архітектура дозволяє гнучко адаптувати систему до різних умов середовища IOT та забезпечує ефективну взаємодію між усіма компонентами, включаючи оператора та служби безпеки. Важливим аспектом є забезпечення масштабованості системи, що дозволяє її інтеграцію в різні IOT-мережі. На діаграмі (рис. 2.1) зображено основні учасники системи та їх взаємодія, а також описані ключові функціональні блоки: сенсори (Sensors), модуль виявлення та класифікації (Detection & Classification), модуль злиття даних (Data Fusion), прийняття рішень (Decision & Control), пом'якшення загроз (Mitigation), інтеграційний модуль (C2 / API) та модуль аудиту (Audit & Compliance). Це дозволяє чітко уявити потоки інформації в системі та призначення кожного елемента.

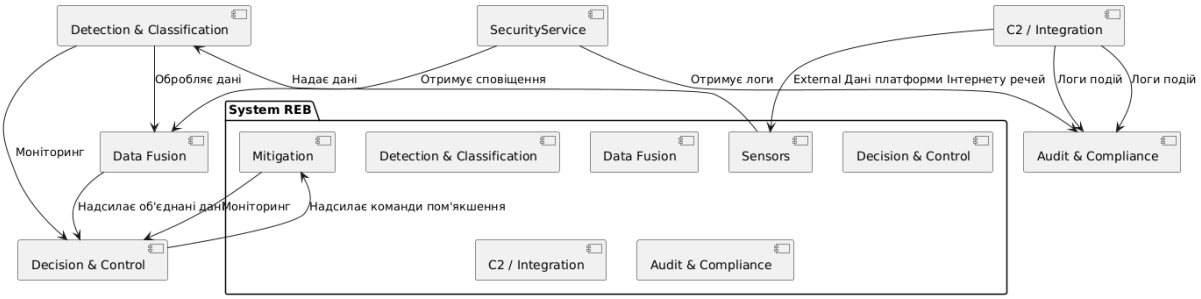


Рисунок 2.1 - Контекстна діаграма

На рис. 2.2 показано внутрішню структуру системи та взаємозв'язок між її модулями. Сенсорні блоки (RF Sensors, Network Monitors, Optical/IR)

передають дані на попередню обробку (Preprocessing), після чого ці дані надходять до модуля виявлення та класифікації (Detection & Classification), який обробляє і класифікує підозрілі об'єкти. Далі, ці дані інтегруються у модуль злиття даних (Data Fusion), щоб сформувати повну картину ситуації. Рішення щодо реагування на загрози приймає модуль прийняття рішень (Decision Engine), який взаємодіє з оператором через інтерфейс користувача (HMI) і контролює модуль пом'якшення загроз (Mitigation). Інтеграційний пакет (C2 / API, Logging / SIEM, Audit & Compliance) забезпечує обмін даними із зовнішніми системами та ведення журналів подій для аудиту. Ця діаграма дозволяє чітко зрозуміти логіку внутрішніх потоків даних і взаємодію компонентів системи.

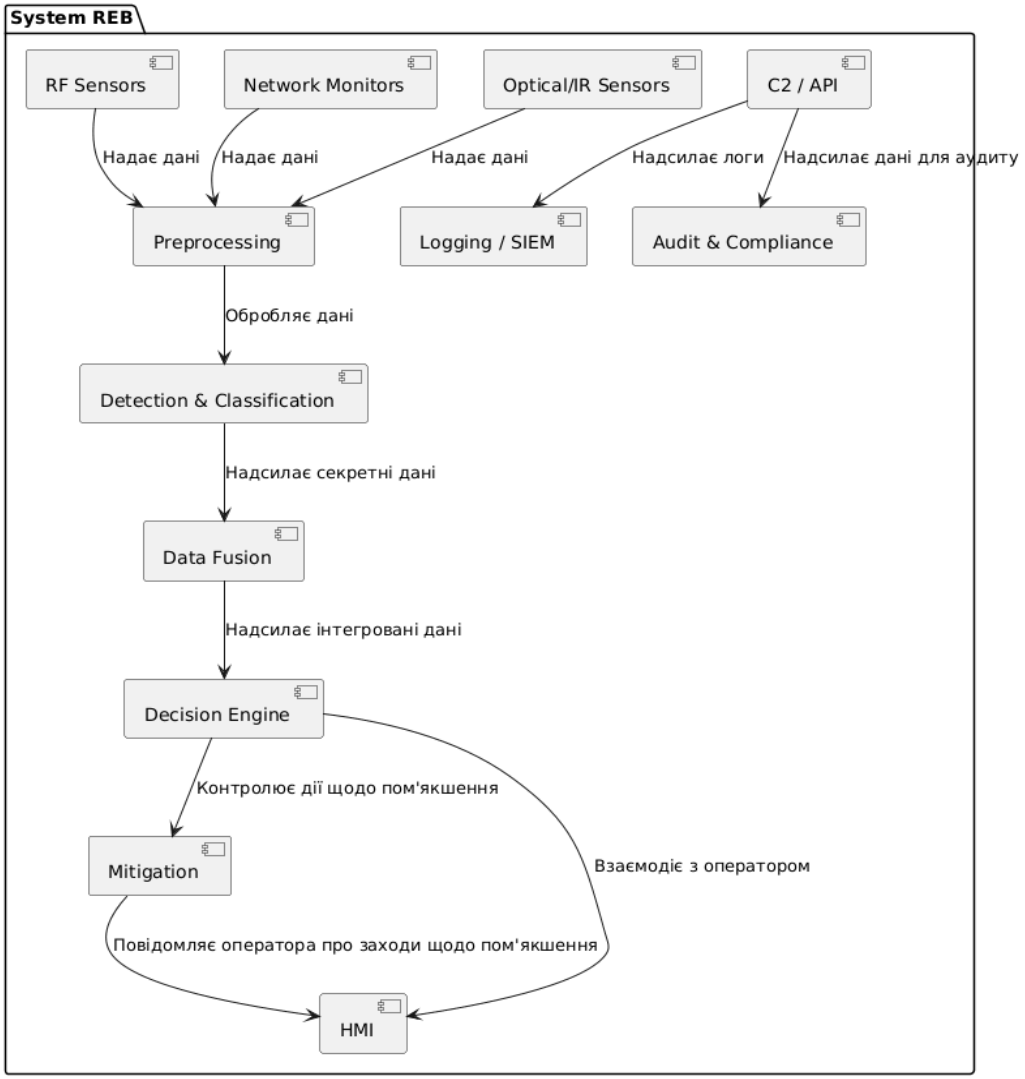


Рисунок 2.2 - Компонентна діаграма

На рис. 2.3 показано часовий потік подій при виявленні підозрілого об'єкта. Спочатку сенсор передає сирі дані до модуля Detection, який здійснює їх класифікацію. Потім ці дані інтегруються у модуль Data Fusion для подальшої оцінки контексту та рівня ризику. На основі зібраної інформації, модуль Decision приймає рішення щодо подальших дій і, за потреби, запитує підтвердження у оператора. Після цього запускається модуль Mitigation, який виконує безпечні дії, такі як ініціація відведення БПЛА або прийняття інших заходів, і надає звіт оператору. Ця діаграма відображає процес обробки інформації - від первинного збору даних до прийняття рішень і реакції системи.

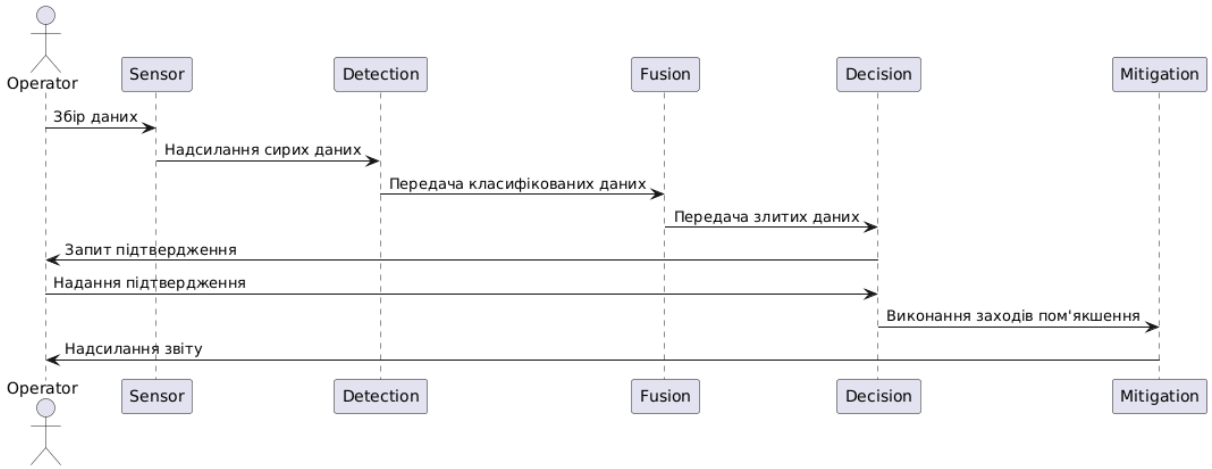


Рисунок 2.3 - Діаграма послідовності

Концептуальна архітектура системи радіоелектронної боротьби в умовах Інтернету речей передбачає модульний та гнучкий підхід до забезпечення безпеки від потенційних загроз з боку безпілотних літальних апаратів. Така архітектура гарантує чітке розмежування функцій, охоплюючи весь процес - від збору даних та їх класифікації до прийняття рішень і забезпечення безпечного реагування на виявлені загрози. Завдяки використанню модулів для злиття даних та прийняття рішень система здатна ефективно інтегрувати інформацію з різноманітних джерел, оцінювати ризики, визначати пріоритети дій та надавати оператору

прозору та зрозумілу взаємодію, що дозволяє швидко реагувати на змінювані умови та загрози, що виникають у мережах IOT.

Підсистема інтеграції надає можливість підключення системи до існуючих IOT-платформ і зовнішніх служб безпеки, що забезпечує масштабованість та ефективну взаємодію з іншими частинами інфраструктури. Це особливо важливо для адаптації системи до різних середовищ і умов. Таким чином, архітектура системи РЕБ поєднує ефективність, безпеку та гнучкість, створюючи основу для подальшого моделювання, проектування та впровадження у реальних умовах. Вона служить концептуальним каркасом, на основі якого можна розробляти алгоритми для виявлення, класифікації та безпечного реагування на загрози з боку БПЛА в мережах IOT.

2.2 Математична модель сигналів БПЛА та пристроїв IOT

Математична модель сигналів є основою для аналізу, моделювання та проектування систем радіоелектронної боротьби в умовах мереж Інтернету речей . Розуміння фізичних характеристик сигналів, які передаються від безпілотних літальних апаратів та IOT-пристроїв, дозволяє створювати алгоритми для виявлення та класифікації цих сигналів у складних умовах електронного середовища.

2.2.1 Модель сигналу БПЛА

Сигнали БПЛА можуть бути представлені у вигляді радіочастотних хвиль, що передаються для цілей телеметрії, команд управління або відеопотоку. Оскільки БПЛА зазвичай працюють на різних частотах, сигнал може бути описаний як комбінація корисного сигналу і шуму. Математично сигнал БПЛА можна представити як дискретний сигнал s(t), який складається з корисного сигналу x(t) та шуму n(t):

$$s(t) = x(t) + n(t)$$

де:

- x(t) – корисний сигнал, який несе інформацію про команду, телеметрію або відеопотік;
- n(t) – шум середовища, який можна моделювати як гаусівський процес або як суму різних джерел інтерференції.

Для цифрових сигналів часто використовуються модуляції типу Amplitude Shift Keying (ASK), Frequency Shift Keying (FSK) або Quadrature Amplitude Modulation (QAM)[18]. Такі сигнали можна представити в дискретній формі як символи

$$s[k] = x[k] + n[k]$$

2.2.2 Модель сигналу IOT-пристрою

Сигнали IOT-пристроїв зазвичай передаються в рамках специфікацій для низькочастотних та малопотужних бездротових технологій, таких як Wi-Fi, ZigBee, LoRa або NB-IOT[19]. Сигнал таких пристроїв має короткі часові інтервали передачі, низьку потужність та часто періодичний характер. Оскільки IOT-пристрої використовуються для збору та передачі сенсорних даних, їхні сигнали часто містять лише короткі пакети даних.

Сигнали IOT-пристроїв зазвичай передаються у вигляді пакета даних з низькою енергоспоживаністю. Їхній сигнал i(t) можна записати аналогічно до сигналу БПЛА:

$$i(t) = y(t) + n(t)$$

де:

- y(t) – корисний сигнал IOT-пристрою;
- n(t) – шум та інтерференція від інших пристроїв у спектрі.

Особливості сигналів IOT-пристроїв включають:

- короткі часові інтервали передачі, де сигнали передаються через невеликі періоди часу;

- низька потужність, що знижує ймовірність перешкод, але також обмежує дальність передачі;
- перерви у передачі сигналів (наприклад, датчики температури можуть передавати дані кожні 10 секунд).

Для аналізу сигналів ІОТ важливо враховувати сумісність спектру сигналів БПЛА та ІОТ, оскільки перехресні інтерференції можуть ускладнювати виявлення загроз, якщо сигнали з різних джерел перекриваються на тих самих частотах.

2.2.3 Модель прийому сигналу

Сигнал, який приймається сенсорами РЕБ, є сумарним ефектом всіх джерел, що працюють в радіочастотному спектрі. Це включає сигнали БПЛА, ІОТ-пристроїв та шум, що створюється іншими пристроями або фоновими перешкодами. Математичну модель сигналу, прийнятого сенсорами РЕБ r(t), можна записати як:

$$r(t) = s(t) + i(t) + n\_env(t)$$

де:

- s(t) – сигнал БПЛА;
- i(t) – сумарний сигнал ІОТ-пристроїв;
- n_env(t) – фоновий шум середовища.

Дана модель дозволяє описати процес виявлення сигналів, а також виділення корисних компонентів через фільтрацію та спектральний аналіз. Процес виявлення в системі РЕБ полягає у вимірюванні параметрів r(t), фільтрації шуму та ідентифікації потенційних загроз.

Дана модель дозволяє описати процес виявлення:

1. Вимірюються параметри r(t);
2. Виконується виділення корисних компонентів (фільтрація, спектральний аналіз);
3. Визначається наявність потенційної загрози та її характеристики.

Таблиця 2.1 - Основні характеристики сигналів БПЛА та IOT-пристроїв

Характеристика	Сигнали БПЛА	Сигнали IOT-пристроїв
Частотний діапазон	2,4–5,8 ГГц (Wi-Fi, FPV)	0,9–2,4 ГГц, LoRa, NB-IOT
Потужність сигналу	Середня/висока	Низька
Тип переданих даних	Телеметрія, команди, відео	Сенсорні дані, короткі пакети
Модуляція	ASK, FSK, QAM	FSK, LoRa Chirp Spread Spectrum
Тривалість передачі	Постійна або періодична	Короткі імпульси, періодичні
Інтерференційна чутливість	Середня/висока	Висока
Особливості	Може переміщатися на великі відстані	Невисока дальність, низька енергоспоживаність

Табл.2.1 показує основні відмінності між сигналами БПЛА та IOT-пристроїв. Це дозволяє оцінити, які параметри сигналу варто використовувати для виявлення потенційних загроз, а також як планувати фільтрацію та класифікацію сигналів у системі РЕБ.

2.2.4 Узагальнена дискретна модель

Для реалізації в цифрових алгоритмах обробки сигналів часто використовують дискретизацію. Дискретну модель сигналу можна виразити через дискретні значення:

$$r[k] = s[k] + i[k] + n[k]$$

де k – номер дискретного відліку, а s[k], i[k], n[k] – дискретні значення відповідних сигналів.

На основі цієї моделі можна будувати алгоритми:

- спектрального аналізу;
- виявлення аномалій у потоці даних;
- класифікації типів об’єктів (БПЛА або IOT-пристрої).

Математична модель сигналів дозволяє прогнозувати характеристики сигналів у різних умовах, оцінювати співіснування сигналів БПЛА та IOT у спектрі, формалізувати процеси виявлення та класифікації, забезпечити основу для побудови алгоритмів РЕБ у симуляціях та реальних тестах. Вона є ключовою ланкою між фізичним рівнем сигналів і логікою системи РЕБ, дозволяючи ефективно інтегрувати модулі спостереження, обробки та прийняття рішень.

2.3 Методи виявлення, класифікації та нейтралізації сигналів

Виявлення сигналів є першочерговим етапом роботи системи радіоелектронної боротьби. Його мета полягає у фіксації наявності потенційно небезпечних сигналів у спектрі та визначенні їхніх характеристик для подальшої обробки. Для виявлення сигналів використовуються методи аналізу енергетичного розподілу сигналу в частотній області, які дозволяють виявити пікові частоти і таким чином відрізнити сигнали БПЛА від фонових сигналів IOT-пристроїв. Спектральний аналіз є важливим інструментом, що дозволяє відокремити сигнали, що мають високі енергетичні характеристики, від тих, що йдуть у фоновому шумі.

Крім спектрального аналізу, важливе значення має часовий аналіз. Цей метод враховує періодичність передачі пакетів, тривалість імпульсів та інші часові характеристики сигналу, що дозволяє точніше ідентифікувати короткі пакети даних IOT-пристроїв або постійні потоки

даних від БПЛА. Для класифікації типів сигналів може бути застосовано методи виявлення аномалій, що базуються на статистичному та машинному аналізі даних. Ці методи дозволяють виявляти сигнали, що не відповідають звичному поведінковому патерну мережі, і таким чином допомагають ідентифікувати потенційно небезпечні об’єкти.

Параметричні методи, які враховують частоту несучої, ширину спектра та форму модуляції сигналу, дають змогу ефективно розрізняти сигнали БПЛА і легітимні IOT-пристрої. Вони дозволяють точно визначити характеристики сигналу, що важливо для подальшої класифікації та нейтралізації загроз.

Класифікація сигналів є наступним етапом після виявлення. На цьому етапі система повинна ідентифікувати джерело сигналу та оцінити його потенційний ризик для безпеки. Після первинного виявлення сигналу система порівнює його характеристики з відомими сигнатурними шаблонами, що дозволяє точно визначити тип пристрою. Одним з основних методів класифікації є сигнатурний аналіз, що включає порівняння спектральних характеристик сигналу з базою даних відомих сигналів. Крім того, в класифікації сигналів використовуються методи машинного навчання, зокрема дерева рішень, нейронні мережі та випадкові ліси, які дозволяють автоматично розпізнавати сигнали навіть в умовах високого рівня шуму або перешкод. Сучасні алгоритми також застосовують поведінковий аналіз, що вивчає час і простір передачі сигналів. Це дозволяє виявляти характерні патерни поведінки, наприклад, траєкторії руху БПЛА або періодичність пакетів IOT-пристроїв, що значно підвищує точність класифікації.

Нейтралізація сигналів є завершальним етапом роботи системи РЕБ. Це важливий процес, що полягає в безпечному реагуванні на потенційно небезпечні об'єкти, з метою мінімізації загроз для безпеки. Основними підходами до нейтралізації є автоматичне або інформаційне сповіщення операторів про виявлену загрозу, що дає можливість здійснити оперативне

реагування. Інші методи включають застосування легітимних обмежень доступу до мережі та ініціацію безпечних протоколів реагування, таких як посадка БПЛА у безпечній зоні або переведення пристроїв ІОТ в режим ізоляції. Крім того, важливим аспектом є аудит і логування всіх дій системи, що забезпечує прозорість операцій і контроль відповідності до законодавчих вимог. Цей механізм є критично важливим для забезпечення правомірного функціонування системи РЕБ в умовах цивільних та корпоративних мереж ІОТ. Концептуальні методи нейтралізації на рівні спектра включають моніторинг частотних діапазонів та оцінку джерел перешкод, що дозволяє забезпечити ефективне реагування без порушення правових норм.

Таблиця 2.2 - Методи виявлення, класифікації та нейтралізації сигналів

Етап системи	Основні методи	Типи сигналів	Коментар
Виявлення	Спектральний аналіз, часовий аналіз, аномалії, параметричні методи	БПЛА, ІОТ	Первинне виявлення сигналу у спектрі
Класифікація	Сигнатурний аналіз, спектральне порівняння, машинне навчання, поведінковий аналіз	БПЛА, ІОТ	Визначення типу джерела та оцінка ризику
Нейтралізація	Інформаційне оповіщення, легітимні обмеження, автоматичне реагування, аудит, концептуальні методи спектра	БПЛА, ІОТ	Безпечне реагування на загрозу, контроль дій

Діаграма діяльності відображає процес роботи системи РЕБ, починаючи з етапу збору сигналів і закінчуючи їх нейтралізацією. Спочатку оператор ініціює збір сигналів, отриманих від різних сенсорів. Це перший крок, коли система починає працювати з даними для подальшої обробки. Після збору сигналів відбувається етап їх виявлення, на якому система проводить попередній аналіз для виявлення потенційно небезпечних об'єктів. Якщо сигнал визначається як небезпечний, система переходить до етапу класифікації сигналу. На цьому етапі сигнал порівнюється з відомими шаблонами, що дозволяє точно визначити тип об'єкта.

Якщо тип сигналу виявляється БПЛА, система передає дані на наступний етап злиття, де відбувається інтеграція сигналів від різних сенсорів для отримання повної картини ситуації. Злиті дані передаються до модуля прийняття рішень, де визначається, чи потребує сигнал подальшої обробки або дій. У разі підтвердження загрози система може запитати у оператора підтвердження для прийняття рішення. Після підтвердження загрози оператором система переходить до нейтралізації сигналу. Це може включати різні заходи, такі як відведення БПЛА в безпечну зону або блокування його сигналу. Після виконання всіх заходів з нейтралізації, оператор отримує звіт про дії системи і статус загрози. Діаграма діяльності показує, як система РЕБ проходить всі етапи - від збору і виявлення сигналів до їх класифікації, прийняття рішень і виконання безпечних заходів реагування на загрози.

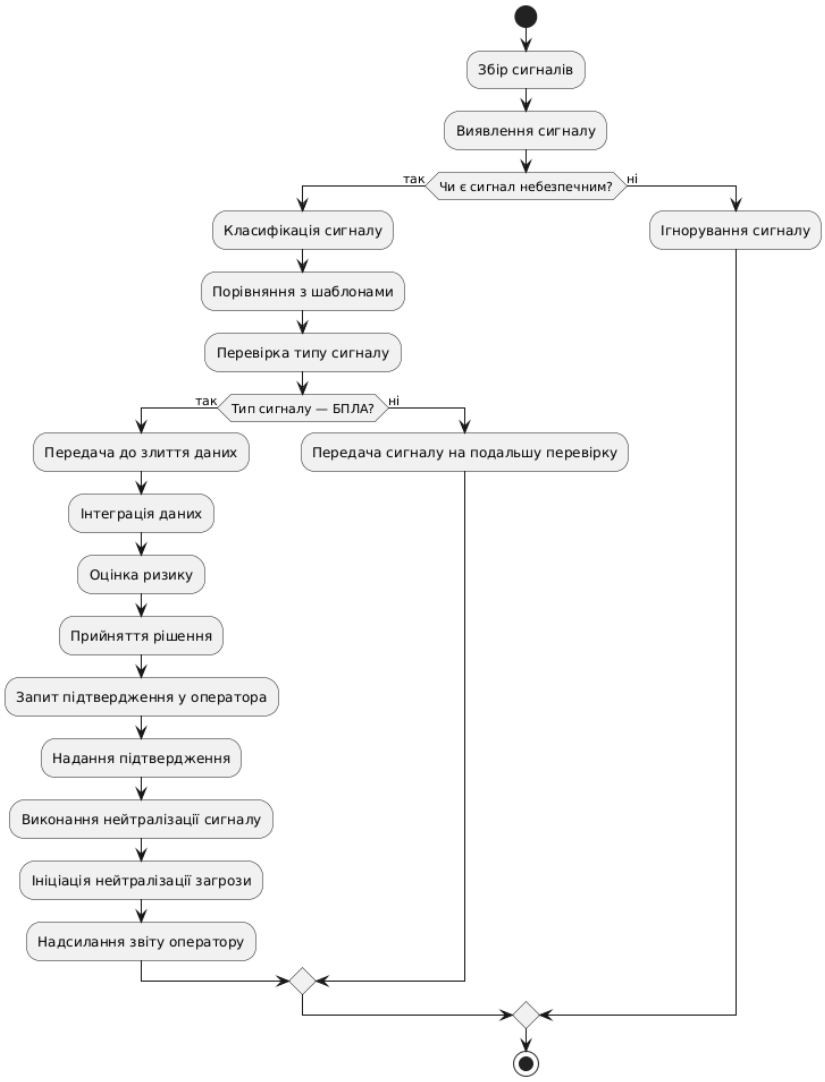


Рисунок 2.4 - Схема процесу роботи системи

Рис. 2.4 показує повний цикл роботи системи РЕБ: від первинного виявлення сигналів до класифікації та безпечної реакції, з можливістю моніторингу і контролю на кожному етапі. Вона інтегрує всі описані методи в єдину концептуальну модель, яка дозволяє ефективно реагувати на загрози в реальному часі.

2.4 Висновки до розділу 2

У другому розділі було детально розглянуто концептуальну архітектуру системи радіоелектронної боротьби в умовах IOT та математичні моделі сигналів БПЛА та IOT-пристроїв. Підхід, заснований

на модульній архітектурі, дозволяє ефективно виявляти, класифікувати та нейтралізувати потенційні загрози, зокрема від БПЛА, в умовах складного електронного середовища, яке характерне для мереж IOT. Ключові компоненти системи, такі як модулі виявлення, класифікації, злиття даних та прийняття рішень, забезпечують гнучкість і масштабованість, що дозволяє адаптувати систему до змінних умов і типів загроз.

Математичні моделі сигналів, розглянуті в цьому розділі, служать основою для розробки алгоритмів виявлення і класифікації сигналів. Моделювання сигналів БПЛА та IOT-пристроїв дозволяє точніше оцінювати їхні характеристики, враховуючи специфіку частотних діапазонів, потужності сигналу, а також взаємодію між ними в загальному електронному середовищі. Зокрема, математичні моделі сигналів БПЛА і IOT-пристроїв надають важливу інформацію для розробки ефективних методів спектрального та часовго аналізу, що допомагають точно виявляти і класифікувати сигнали, а також забезпечують можливість відрізняти загрози від легітимних пристроїв. Врахування параметричних методів, таких як частота несучої та форма модуляції, дозволяє більш ефективно розподіляти ресурси системи РЕБ і точно реагувати на сигнали від різних типів пристроїв.

Особливу увагу було приділено методам виявлення, класифікації та нейтралізації сигналів. Використання спектрального аналізу, методів машинного навчання та аналізу поведінкових патернів дозволяє значно покращити точність виявлення загроз та класифікації сигналів навіть в умовах високого рівня шуму. Завдяки цьому система РЕБ здатна не тільки виявляти загрози, але й точно оцінювати їхній ризик для подальшого безпечного реагування.

В результаті, розроблена концептуальна архітектура та математичні моделі сигналів стають основою для створення інтегрованої системи, яка здатна ефективно працювати в реальному часі, забезпечуючи високу точність виявлення та нейтралізації загроз, що виникають у мережах IOT.

Таким чином, система РЕБ у умовах ІОТ є важливим інструментом для забезпечення безпеки в умовах сучасних загроз, включаючи БПЛА.

3 РОЗРОБКА ПРОГРАМНОГО ЗАСОБУ ДЛЯ ЕМУЛЯЦІЇ ТЕХНОЛОГІЙ РЕБ

3.1 Загальна структура та логіка Python-програми

Python-програма для реалізації системи радіоелектронної боротьби в умовах IOT побудована з кількох модулів, кожен з яких виконує специфічну функцію, що є частиною загальної логіки виявлення та нейтралізації загроз. У цій частині описано загальну структуру програми, функціональність кожного з модулів, а також їх взаємодія між собою для ефективної роботи системи.

Програма складається з кількох основних класів (модулів), що реалізують окремі підсистеми системи РЕБ. Кожен з модулів має чітко визначену функціональність, зокрема:

1. Підсистема спостереження (Sensors): здійснює збір даних з різних джерел, таких як радіочастотні сенсори, мережеві монітори IOT, оптичні та інфрачервоні сенсори. Ці сенсори відповідальні за реальний моніторинг середовища та виявлення підозрілих об'єктів у межах зони спостереження. В програмі ця підсистема реалізована класом Sensors, який має методи для генерування та обробки сигналів.
2. Підсистема виявлення та класифікації (Detection & Classification): цей модуль займається обробкою зібраних даних, виявленням потенційно небезпечних об'єктів та їх класифікацією. Він відповідає за спектральний аналіз сигналів, а також за екстракцію основних характеристик сигналів (наприклад, амплітуд). Клас DetectionClassification реалізує ці функції, включаючи спектральний аналіз (FFT) та методи екстракції характеристик сигналів.
3. Модуль злиття даних (Data Fusion): цей модуль інтегрує інформацію з різних сенсорів для створення єдиної повної картини ситуації, що дозволяє врахувати різноманітні джерела даних, з'єднуючи їх в

єдиний контекст для подальшої обробки. Клас DataFusion виконує обчислення середніх значень або інших методів інтеграції для злиття даних.

4. Модуль пом'якшення загроз (Mitigation): відповідає за реалізацію заходів з нейтралізації загрози після того, як вона була ідентифікована і класифікована. Це дії, такі як відведення БПЛА або блокування сигналу IOT. Клас Mitigation здійснює управління цими діями в залежності від типу загрози.
5. Підсистема управління та інтеграції (C2 / Integration): цей модуль відповідає за взаємодію з іншими системами, платформами IOT і зовнішніми інформаційними системами. Він також відповідає за логування подій і аудит для подальшого аналізу та звітності. Клас C2Integration дозволяє здійснювати моніторинг системи та зберігати необхідні дані.

Програма працює в реальному часі, де кожен з компонентів здійснює своєчасне збирання даних, їх обробку, класифікацію, а також прийняття рішень про подальші дії. Програма починається з ініціалізації підсистеми спостереження. Вона генерує сигнали для різних типів пристроїв (наприклад, БПЛА чи IOT-пристрої) за допомогою методу generate_signal класу Sensors. Це забезпечує збирання даних у реальному часі від сенсорів, які можуть бути радіочастотними, оптичними або іншими типами датчиків.

Зібрані сигнали передаються в підсистему виявлення та класифікації. Клас DetectionClassification здійснює спектральний аналіз сигналів і витягує ключові характеристики сигналів для подальшої обробки. Це дозволяє не тільки виявляти потенційно небезпечні сигнали, але й класифікувати їх на основі їхніх характеристик (наприклад, відрізнити БПЛА від IOT-пристроїв). Потім, зібрані та класифіковані дані об'єднуються за допомогою класу DataFusion, що дозволяє створити єдину

картину ситуації, враховуючи дані з різних сенсорів. Це важливо для точнішої оцінки загроз та правильного реагування на них.

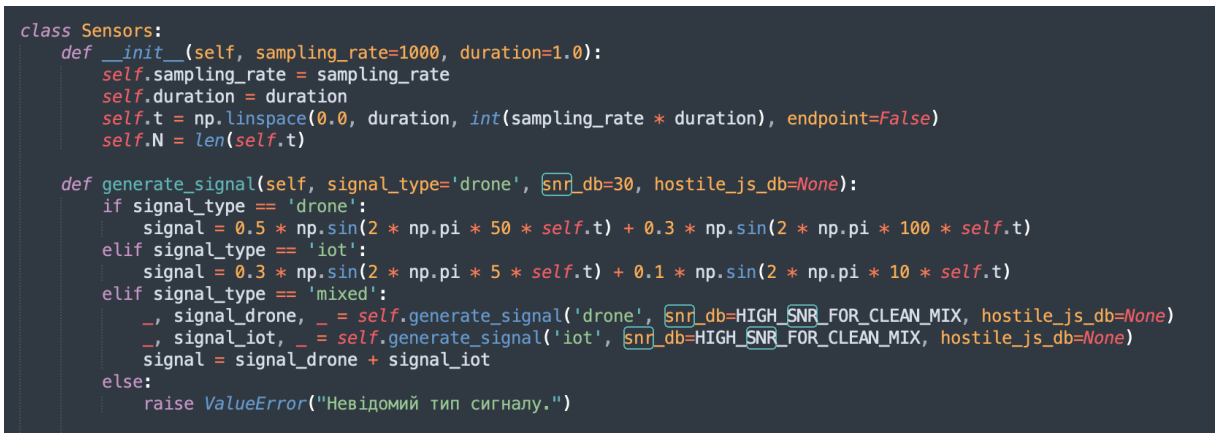
На основі злитих даних система визначає, чи є сигнал небезпечним. Клас DecisionControl приймає рішення, чи потрібно вжити заходів для нейтралізації загрози. У випадку підтвердження загрози система запитує підтвердження у оператора або автоматично ініціює процес нейтралізації. Якщо загроза підтверджена, клас Mitigation ініціює безпечні дії, наприклад, відведення БПЛА або переведення IOT-пристроїв в режим ізоляції. Це забезпечує оперативне реагування на потенційно небезпечні ситуації.

Підсистема управління та інтеграції (C2Integration) забезпечує взаємодію з іншими зовнішніми системами та платформами IOT. Крім того, цей компонент відповідає за логування подій та збереження даних для подальшого аналізу та звітності, що є важливим для забезпечення прозорості і відповідності законодавчим вимогам.

Програма побудована таким чином, що кожен модуль виконує свою чітко визначену функцію в рамках загальної системи РЕБ. Це дозволяє ефективно виявляти, класифікувати та нейтралізувати загрози в реальному часі, забезпечуючи високий рівень безпеки для мереж IOT. Комбінація збору даних, аналізу, прийняття рішень та нейтралізації загроз дозволяє цій системі працювати на найвищому рівні ефективності, реагуючи на різноманітні загрози у сучасних умовах цифрової безпеки.

3.2 Реалізація модулів генерації, обробки та класифікації сигналів

У програмі реалізовано кілька модулів, кожен з яких виконує певні функції у рамках системи радіоелектронної боротьби. Нижче наведено опис кожного з цих модулів, а також їхніх функцій та параметрів.



```python
class Sensors:
    def __init__(self, sampling_rate=1000, duration=1.0):
        self.sampling_rate = sampling_rate
        self.duration = duration
        self.t = np.linspace(0.0, duration, int(sampling_rate * duration), endpoint=False)
        self.N = len(self.t)

    def generate_signal(self, signal_type='drone', snr_db=30, hostile_js_db=None):
        if signal_type == 'drone':
            signal = 0.5 * np.sin(2 * np.pi * 50 * self.t) + 0.3 * np.sin(2 * np.pi * 100 * self.t)
        elif signal_type == 'iot':
            signal = 0.3 * np.sin(2 * np.pi * 5 * self.t) + 0.1 * np.sin(2 * np.pi * 10 * self.t)
        elif signal_type == 'mixed':
            _, signal_drone, _ = self.generate_signal('drone', snr_db=HIGH_SNR_FOR_CLEAN_MIX, hostile_js_db=None)
            _, signal_iot, _ = self.generate_signal('iot', snr_db=HIGH_SNR_FOR_CLEAN_MIX, hostile_js_db=None)
            signal = signal_drone + signal_iot
        else:
            raise ValueError("Невідомий тип сигналу.")
```

Рисунок 3.1 - Модуль спостереження (Sensors)

Модуль Sensors є частиною симулятора, який використовується для генерування радіосигналів та додавання до них шуму та ворожої завади. Клас має методи для створення корисного сигналу різних типів (наприклад, для дронів або IOT-пристроїв), а також можливість додавати природний шум та ворожу заваду. Клас розраховує потужність сигналу, шуму та завади, а також обчислює показник SINR (Signal-to-Interference-plus-Noise Ratio), який характеризує якість сигналу після додавання завад. В залежності від типу сигналу (наприклад, дрон або IOT) генерується відповідний сигнал, а додавання шуму та завади дозволяє моделювати реальні умови, з якими можуть зіткнутися системи радіоелектронної боротьби. У табл.3.1 описано функції модулю.

Таблиця 3.1 Опис функцій модулю спостереження

Функція	Опис	Параметри
__init__(self, sampling_rate=1000, duration=1.0)	Ініціалізує об'єкт класу, задаючи параметри зразка (частоту дискретизації) і тривалість сигналу.	- sampling_rate: частота дискретизації сигналу в Гц (за замовчуванням 1000). - duration: тривалість сигналу в секундах (за замовчуванням 1.0 сек).

Продовження таблиці 3.1

Функція	Опис	Параметри
generate_signal(self, signal_type='drone', snr_db=30, hostile_js_db=None)	Генерує корисний сигнал з шумом та ворожою завадою, а також розраховує фактичний SINR (Signal-to-Interference-plus-Noise Ratio).	- signal_type: тип корисного сигналу ('drone', 'IOT', 'mixed'). - snr_db: відношення сигнал/шум у дБ для природного шуму. - hostile_js_db: відношення ворожої завади до сигналу (J/S) в дБ.
generate_signal('drone')	Генерує сигнал для типу 'drone', що включає синусоїди з частотами 50 Гц і 100 Гц.	Без параметрів (тип сигналу за замовчуванням: 'drone').
generate_signal('IOT')	Генерує сигнал для типу 'IOT', що включає синусоїди з частотами 5 Гц і 10 Гц.	Без параметрів (тип сигналу за замовчуванням: 'IOT').
generate_signal('mixed')	Генерує змішаний сигнал, що складається з сигналів типу 'drone' і 'IOT' при високому SNR, щоб мінімізувати шум.	Без параметрів (тип сигналу за замовчуванням: 'mixed').
signal_power	Розраховує потужність сигналу як дисперсію (варіацію) корисного сигналу, щоб визначити рівень потужності для подальших розрахунків шуму та завади.	- signal: корисний сигнал для обчислення його потужності.

Продовження таблиці 3.1

noise_power	Обчислює потужність природного шуму на основі заданого відношення сигнал/шум (SNR).	- snr_db: відношення сигнал/шум у дБ для шуму.
total_interference	Додає до сигналу природний шум та ворожу заваду, якщо такова задана, і створює загальну заваду для сигналу.	- noise: природний шум. - jamming: ворожа завада, що додається, якщо задано.
sinr_db	Обчислює фактичне відношення сигнал/завада (SINR) в дБ, яке залежить від потужності сигналу та завади (як природного шуму, так і ворожої завади).	- signal_power: потужність сигналу. - total_interference_power: потужність загальної завади (шум + ворожа завада).
final_signal	Обчислює та повертає фінальний сигнал, який є сумою корисного сигналу та завади (як природного шуму, так і ворожої завади).	- signal: корисний сигнал. - total_interference: загальна завада (шум + ворожа завада).

Модуль DetectionClassification відповідає за класифікацію радіосигналів на основі їх спектральних характеристик. Він включає функціонал для аналізу сигналів, екстракції ознак, тренування моделі машинного навчання та передбачення класів сигналів (наприклад, IOT або дрон).

```python
class DetectionClassification:
    def __init__(self):
        self.scaler = StandardScaler()
        self.model = SVC(kernel='linear', C=1.0, probability=True)
        self.labels = {0: 'IoT', 1: 'Drone'}
        self.is_trained = False
        self.training_data = {'X': None, 'y': None}

    def spectral_analysis(self, signal, sampling_rate=1000):
        N = len(signal)
        fft_values = fft(signal)
        freqs = fftfreq(N, 1 / sampling_rate)
        magnitude = np.abs(fft_values) / N
        return freqs[:N // 2], magnitude[:N // 2]

    def extract_features(self, signal, sampling_rate=1000):
        freqs, magnitude = self.spectral_analysis(signal, sampling_rate)

        idx_50 = np.argmin(np.abs(freqs - 50)); feature_amp_50 = magnitude[idx_50]
        idx_5 = np.argmin(np.abs(freqs - 5)); feature_amp_5 = magnitude[idx_5]

        magnitude_norm = magnitude / np.sum(magnitude)
        magnitude_norm_clean = magnitude_norm[magnitude_norm > 1e-10]
        feature_centroid = np.sum(freqs[:len(magnitude_norm)] * magnitude_norm)

        feature_entropy = entropy(magnitude_norm_clean)

        return [feature_amp_50, feature_amp_5, feature_centroid, feature_entropy]

    def train_model(self, X, y):
        self.scaler.fit(X)
        X_scaled = self.scaler.transform(X)
        self.model.fit(X_scaled, y)
        self.is_trained = True
        self.training_data['X'] = X
        self.training_data['y'] = y

    def classify(self, features):
        features_scaled = self.scaler.transform([features])
        prediction = self.model.predict(features_scaled)[0]
        try:
            confidence = self.model.predict_proba(features_scaled)[0][prediction]
        except:
            confidence = 1.0
        return self.labels[prediction], confidence
```

Рисунок 3.2 - Модуль виявлення та класифікації (Detection & Classification)

Клас використовує методи спектрального аналізу для отримання ключових ознак сигналу, таких як амплітуда на певних частотах (наприклад, 50 Гц для дронів і 5 Гц для IOT), спектральний центроїд і спектральну ентропію. Ці ознаки потім використовуються для тренування лінійної підтримувальної векторної машини (SVC), яка застосовується для класифікації нових сигналів.

SVM (Support Vector Machine) - це потужний метод машинного навчання для класифікації та регресії, який активно використовується для вирішення задач розпізнавання образів, класифікації текстів, біоінформатики, фінансового прогнозування та інших задач. В основі SVM лежить ідея побудови гіперплощини (або гіперплощин для нелінійних

даних), яка найкраще розділяє різні класи у просторі ознак. У табл.3.2 описано функції модулю.

Таблиця 3.2 - Опис функцій модулю виявлення та класифікації

Функція	Опис	Параметри
__init__(self)	Ініціалізує об'єкт класу, налаштовуючи моделі для стандартизації та класифікації сигналів (SVC).	Без параметрів.
spectral_analysis(self, signal, sampling_rate=1000)	Виконує спектральний аналіз сигналу, використовуючи FFT (перетворення Фур'є), для отримання амплітудного спектра та частот.	- signal: вхідний сигнал. - sampling_rate: частота дискретизації сигналу. (за замовчуванням 1000 Гц)
extract_features(self, signal, sampling_rate=1000)	Витягує ключові ознаки з сигналу, зокрема амплітуду на ключових частотах (50 Гц для дронів, 5 Гц для IOT), спектральний центроїд і ентропію спектра.	- signal: вхідний сигнал. - sampling_rate: частота дискретизації сигналу. (за замовчуванням 1000 Гц)
train_model(self, X, y)	Навчає модель на основі екстракованих ознак, використовуючи SVC (підтримувальна векторна машина).	- X: матриця ознак (особливостей). - y: мітки класів (0 або 1, IOT або Drone).
classify(self, features)	Класифікує новий сигнал за допомогою навченого SVC і повертає передбачення класу та впевненість.	- features: екстраковані ознаки сигналу, що потрібно класифікувати.

Модуль DataFusion відповідає за об'єднання результатів класифікації і прийняття рішень на основі аналізу характеристик сигналу. Він використовує інформацію про класифікацію (наприклад, чи сигнал є дроновим або IOT), впевненість у класифікації та спектральні характеристики сигналу для визначення рівня загрози та типу необхідного контрзаходу.

```python
class DataFusion:
    def fuse_and_decide(self, classification_result, confidence, features):
        CONFIDENCE_THRESHOLD = 0.85
        DRONE_AMPLITUDE_THRESHOLD = 0.008

        status = 'SAFE'
        threat_type = 'none'
        jamming_type = 'none'

        drone_amp = features[0]
        iot_amp = features[1]

        if confidence >= CONFIDENCE_THRESHOLD:
            if classification_result == 'Drone':
                if drone_amp >= DRONE_AMPLITUDE_THRESHOLD:
                    # ПРІОРИТЕТ: Широкосмугове заглушення (найсильніше)
                    status = 'DANGER'
                    threat_type = 'drone'
                    jamming_type = 'WIDEBAND'
                else:
                    # Drone, але слабкий сигнал - вузькосмуговий РЕБ
                    status = 'WARNING'
                    threat_type = 'drone'
                    jamming_type = 'NARROWBAND'

            elif classification_result == 'IoT':
                if iot_amp > 0.005: # Порогове значення для сильного IoT
                    # ВАРІАНТ: Вузькосмугове заглушення
                    status = 'WARNING'
                    threat_type = 'iot'
                    jamming_type = 'NARROWBAND'
```

Рисунок 3.3 - Модуль злиття даних та прийняття рішень (Data Fusion)

Клас дозволяє приймати рішення, чи варто застосовувати широкий чи вузькосмуговий заглушення на основі потужності сигналу. У табл.3.3 модулю злиття даних та прийняття рішень.

Таблиця 3.3 - Опис функцій модулю злиття даних та прийняття рішень

Функція	Опис	Параметри
fuse_and_decide(self, classification_result, confidence, features)	Приймає рішення на основі результатів класифікації сигналу, його впевненості та ключових характеристик сигналу. Визначає статус системи і тип загрози.	- classification_result: Результат класифікації сигналу ('Drone', 'IOT'). - confidence: Впевненість у класифікації (від 0 до 1). - features: Ознаки сигналу, зокрема амплітуди на частотах для БПЛА та IOT.
status	Статус системи (наприклад, "SAFE", "DANGER", "WARNING", "DANGER_FUSION", "SAFE_LOW_CONFIDENCE").	Значення, яке визначається залежно від порогових значень для впевненості та амплітуд сигналу.
threat_type	Тип загрози, що може бути "none", "drone", "IOT", або "mixed" залежно від класифікації та рівня сигналу.	Змінна, що зберігає тип загрози на основі класифікації та рівня потужності сигналу.
jamming_type	Тип контрзаходу, який потрібно застосувати, може бути "WIDEBAND" або "NARROWBAND" залежно від типу загрози і рівня сигналу.	Змінна, що визначає тип контрзаходу, який потрібно застосувати для зниження ефективності ворожого сигналу.

Модуль пом'якшення загроз генерує заглушення для нейтралізації загрози, якщо це необхідно. Заглушення може бути широкосмуговим або вузькосмуговим в залежності від типу загрози.

```python
class Mitigation:
    def __init__(self, sampling_rate=1000, duration=1.0):
        self.sampling_rate = sampling_rate
        self.t = np.linspace(0.0, duration, int(sampling_rate * duration), endpoint=False)
        self.N = len(self.t)

    def generate_jamming_signal(self, jamming_type, strength=1.0):
        if jamming_type == 'WIDEBAND':
            jamming_power = strength * 5
            jamming_signal = np.random.randn(self.N) * np.sqrt(jamming_power)
        elif jamming_type == 'NARROWBAND':
            target_freq = 50 if strength >= 1.5 else 5 # Умовний вибір частоти
            jamming_power = strength * 3
            jamming_signal = np.sqrt(jamming_power) * np.sin(2 * np.pi * target_freq * self.t)
        else:
            return np.zeros(self.N)

        return jamming_signal
```

Рисунок 3.4 - Модуль пом'якшення загроз (Mitigation)

Модуль також оцінює ефективність застосованого РЕБ шляхом порівняння характеристик сигналів до та після застосування заглушення. У табл.3.4 описано функції модулю.

Таблиця 3.4 - Опис функцій модулю пом'якшення загроз

Функція	Опис	Параметри
__init__(self, sampling_rate=1000, duration=1.0)	Ініціалізує об'єкт класу, визначаючи параметри частоти дискретизації та тривалості сигналу.	- sampling_rate: частота дискретизації сигналу в Гц (за замовчуванням 1000). - duration: тривалість сигналу в секундах (за замовчуванням 1.0).

Продовження таблиці 3.4

Функція	Опис	Параметри
generate_jamming_signal(self, jamming_type, strength=1.0)	Генерує сигнал контрзаходу (РЕБ), який залежить від типу заглушення (широкосмугове або вузькосмугове) і його сили.	- jamming_type: тип заглушення ('WIDEBAND' або 'NARROWBAND'). - strength: сила заглушення (за замовчуванням 1.0).
apply_jamming(self, signal, jamming_signal)	Додає сигнал контрзаходу до отриманого (оригінального) сигналу, утворюючи сигнал з контрзаходом.	- signal: оригінальний сигнал. - jamming_signal: сигнал контрзаходу.
evaluate_jamming_effectiveness(self, original_features, mitigated_signal, detection_module, threat_type_detected)	Оцінює ефективність застосованого РЕБ за зниженням ключових ознак сигналу після застосування контрзаходу.	- original_features: оригінальні ознаки сигналу. - mitigated_signal: сигнал після застосування РЕБ. - detection_module: об'єкт модуля класифікації для екстракції ознак. - threat_type_detected: тип загрози, який був виявлений.

Модуль надає функціонал для створення і застосування контрзаходів (заглушення) для різних типів загроз (дрони або IOT), а також для оцінки ефективності цих заходів на основі спектральних характеристик сигналу. Це важливий інструмент для систем радіоелектронної боротьби.

```
class C2Integration:
    def __init__(self, log_output):
        self.log_output = log_output

    def log_event(self, event):
        self.log_output.insert(tk.END, f"{event}\n")
        self.log_output.see(tk.END)
```

Рисунок 3.5 - Підсистема управління та інтеграції (C2 / Integration)

Модуль управління та інтеграції забезпечує логування подій у системі та взаємодію з іншими платформами. Він дозволяє зберігати журнал подій, що відбуваються під час роботи системи. У табл.3.6 описано функції підсистеми управління та інтеграції.

Таблиця 3.5 - Опис функцій підсистеми управління та інтеграції

Функція	Опис	Параметри
__init__(self, log_output)	Ініціалізує об'єкт класу та задає текстове поле, куди будуть виводитися повідомлення.	- log_output: текстове поле (ScrolledText), куди будуть додаватися логовані події.
log_event(self, event)	Додає нове повідомлення (подію) до текстового поля і автоматично прокручує вікно, щоб показати останнє повідомлення.	- event: повідомлення, яке потрібно вивести в текстове поле (наприклад, повідомлення про стан системи).

Кожен з модулів має чітко визначені функції, що забезпечують виконання конкретних завдань у системі РЕБ. Параметри для кожної функції забезпечують гнучкість налаштувань, а результати цих функцій використовуються для подальшої обробки і прийняття рішень. Модуль спостереження відповідає за збір і генерацію сигналів. Модуль виявлення та класифікації аналізує ці сигнали, екстрагує важливі характеристики і класифікує їх. Модуль злиття даних інтегрує дані з різних джерел,

створюючи єдину картину ситуації. Модуль прийняття рішень на основі класифікації та характеристик сигналу приймає рішення щодо застосування РЕБ. Модуль пом'якшення загроз генерує заглушення для нейтралізації загрози, якщо необхідно. Підсистема управління та інтеграції забезпечує логування подій і інтеграцію з іншими платформами.

3.3 Емуляція процесів виявлення та нейтралізації у віртуальній IOT-мережі

Віртуальна IOT-мережа, в контексті радіоелектронної боротьби, включає в себе моделювання процесів виявлення і нейтралізації загроз, таких як дрони або пристрої Інтернету речей, з використанням різних технологій виявлення та протидії завадам.

Модуль Sensors генерує різні типи радіосигналів, які використовуються для моделювання роботи IOT-мережі та виявлення загроз. Для кожного типу сигналу (дрон, IOT, змішаний) генерація сигналу включає додавання природного шуму та, при необхідності, ворожої завади. Це дозволяє моделювати реальні умови, з якими стикаються системи радіоелектронної боротьби в реальних умовах.

Сигнали, згенеровані у Sensors, потім проходять через модуль DetectionClassification, який використовує методи спектрального аналізу для екстракції ознак сигналу, таких як амплітуда на конкретних частотах (50 Гц для дронів та 5 Гц для IOT), спектральний центроїд та ентропія спектра. Класифікація здійснюється за допомогою лінійної підтримувальної векторної машини, що дозволяє визначити, чи є сигнал від дрону або IOT-пристрою. Модель класифікації тренується на різних рівнях сигнал/шум. Після класифікації, результати передаються в DataFusion, де на основі впевненості в класифікації та значень ключових ознак (амплітуди на частотах) приймається рішення про тип загрози та необхідність застосування контрзаходів. Якщо впевненість висока і загроза

класифікована як дрон, система може прийняти рішення про застосування широкосмугового заглушення (WIDEBAND) або вузькосмугового (NARROWBAND) залежно від амплітуди сигналу.

Модуль Mitigation генерує сигнали контрзаходів, які можуть бути як широкосмуговими, так і вузькосмуговими. Якщо на основі рішення виявлено, що сигнал потребує нейтралізації, до сигналу додається сигнал контрзаходу. Після застосування контрзаходу ефективність цих заходів оцінюється шляхом порівняння ознак сигналу до та після застосування РЕБ. Модуль C2Integration використовується для виведення результатів на інтерфейс користувача. Він логує події, такі як процес навчання моделі, результати класифікації, застосування контрзаходів та зміни в ознаках сигналів. Це забезпечує зручний моніторинг та аналіз роботи системи в реальному часі. Користувач може відстежувати хід симуляцій за допомогою графіків, що відображають як сигнал до, так і після застосування РЕБ. Візуалізація надає інформацію про амплітуду сигналу, спектр, ефективність контрзаходів та зміни в ключових ознаках сигналу.

Всі процеси контролюються через графічний інтерфейс користувача, створений за допомогою бібліотеки Tkinter. Користувач може налаштувати параметри, такі як рівень сигналу/шуму (SNR), рівень ворожої завади (J/S), силу контрзаходів, а також запускати симуляції для різних типів загроз (дрони, IOT, змішана загроза). Результати кожної симуляції, а також графіки, що показують ефективність застосованих контрзаходів, виводяться на екран.

Віртуальна IOT-мережа в цьому симуляторі моделює реальні умови радіоелектронної боротьби, дозволяючи досліджувати виявлення загроз та нейтралізацію сигналів у складних умовах з шумом та завадами. Цей процес включає збирання даних, класифікацію, прийняття рішень щодо застосування контрзаходів та моніторинг ефективності контрзаходів через візуалізацію та логування подій.

3.4 Висновки до розділу 3

У третьому розділі було розглянуто структуру програмного забезпечення для емулювання технологій радіоелектронної боротьби в умовах ІОТ. Програма побудована на основі кількох основних модулів, кожен з яких виконує свою чітко визначену функцію для забезпечення ефективної роботи системи РЕБ. Всі модулі взаємодіють між собою, що дозволяє забезпечити високий рівень інтеграції та взаємодії в рамках системи. Модуль спостереження генерує різноманітні типи сигналів, що імітують реальні умови роботи БПЛА та ІОТ-пристроїв. До цих сигналів додається шум і ворожа завада, що моделює реальні умови роботи систем РЕБ.

Подальша обробка зібраних сигналів здійснюється за допомогою модуля виявлення та класифікації, який здійснює спектральний аналіз сигналів, виділяє ключові ознаки та класифікує сигнали як "дрон" або "ІОТ-пристрій". Результати класифікації передаються в модуль злиття даних, який об'єднує інформацію з різних джерел і приймає рішення щодо рівня загрози та необхідності застосування контрзаходів.

Якщо загроза підтверджена, модуль пом'якшення загроз генерує відповідні сигнали контрзаходів, які можуть бути широкосмуговими або вузькосмуговими в залежності від типу загрози. Модуль управління та інтеграції забезпечує логування подій, моніторинг ефективності системи та взаємодію з іншими платформами ІОТ для подальшого аналізу та прийняття рішень.

Таким чином, розроблена система дозволяє не лише ефективно виявляти та класифікувати загрози, але й здійснювати оперативну нейтралізацію через застосування різних контрзаходів. Всі модулі працюють у реальному часі, що забезпечує високий рівень безпеки для мереж ІОТ в умовах сучасних загроз.

4 ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ

4.1 Опис експериментальних сценаріїв і параметрів моделювання

У цьому підрозділі описано експериментальні сценарії та параметри, які використовуються для тестування та моделювання роботи системи радіоелектронної боротьби в умовах IOT. Програма, що реалізує ці технології, має на меті моделювати реальні умови боротьби з БПЛА та IOT-пристроями, використовуючи різні типи сигналів, шуму та ворожої завади. Програма побудована на основі кількох основних модулів, кожен з яких виконує специфічну функцію в рамках загальної системи. Зокрема, модулі Sensors, DetectionClassification, DataFusion, Mitigation та C2Integration дозволяють проводити симуляцію процесів виявлення, класифікації та нейтралізації загроз у реальному часі.

Програма генерує сигнали для різних типів пристроїв, таких як БПЛА та IOT-пристрої. Для цього використовуються синусоїдальні хвилі з різними частотами для кожного типу пристрою. Модуль Sensors відповідає за генерацію таких сигналів:

- БПЛА: використовує частоти 50 Гц та 100 Гц.
- IOT: використовує частоти 5 Гц та 10 Гц.
- змішаний сигнал: поєднання сигналів БПЛА та IOT, що дозволяє моделювати ситуації, коли в зоні спостереження одночасно працюють різні пристрої.

Для імітації реальних умов, до кожного сигналу додається природний шум та можливі перешкоди від ворожої завади. Програма дозволяє змінювати рівень сигнал/шум (SNR) у діапазоні від 5 до 50 дБ. Окрім цього, система підтримує введення ворожої завади (hostile jamming) в діапазоні від -10 до 20 дБ, що моделює вплив зовнішніх джерел перешкод.

Для виявлення та класифікації сигналів використовується спектральний аналіз, що дозволяє виділяти ключові характеристики

сигналу, такі як амплітуда на специфічних частотах, спектральний центроїд і спектральну ентропію. Програма реалізує методи машинного навчання (підтримувальна векторна машина, SVC), щоб класифікувати сигнали як "IOT" або "Drone" на основі цих характеристик.

Програма передбачає налаштування кількох параметрів:

1. Рівень сигнал/шум (SNR) визначає співвідношення між корисним сигналом і фоновим шумом.
2. Рівень ворожої завади (J/S), який додається до сигналу для моделювання реальних умов.
3. Сила контрзаходу дозволяє користувачу регулювати потужність застосовуваних засобів РЕБ.

Користувач може налаштувати ці параметри через графічний інтерфейс, який дозволяє запускати симуляції для різних типів загроз (дрони, IOT-пристрої, змішана загроза). Результати симуляцій виводяться на екран у вигляді графіків, що показують ефективність застосованих контрзаходів і зміну характеристик сигналів.

Для повного тестування програмного засобу радіоелектронної боротьби було розроблено декілька тестових сценаріїв, що охоплюють всі функціональні можливості програми. Кожен тестовий сценарій спрямований на перевірку різних аспектів роботи системи, таких як генерація сигналів, виявлення та класифікація загроз, злиття даних, застосування контрзаходів та інтеграція з іншими системами. Тестування проводиться для різних умов: змінний рівень сигналу/шуму (SNR), рівень ворожої завади (J/S), а також для різних типів загроз.

Сценарій №1. Генерація сигналів та вплив шуму і завади

Мета: перевірити правильність генерації сигналів для різних типів пристроїв (БПЛА, ІОТ, змішані сигнали) і вплив природного шуму та ворожої завади на ці сигнали.

Дії:

1. Генерація сигналів:

1. Генерація сигналу для БПЛА з рівнем SNR 30 дБ.
2. Генерація сигналу для ІОТ-пристрою з рівнем SNR 30 дБ.
3. Генерація змішаного сигналу з високим SNR (наприклад, 60 дБ).

2. Додавання шуму та завади:

1. Додавання природного шуму до сигналу для БПЛА з рівнем SNR 20 дБ та ворожої завади J/S = 5 дБ.
2. Додавання ворожої завади до сигналу ІОТ з рівнем SNR 30 дБ без додавання ворожої завади.
3. Генерація змішаного сигналу з рівнем SNR 40 дБ та ворожою завадею J/S = 10 дБ.

Очікуваний результат:

1. Сигнали для БПЛА та ІОТ повинні бути коректно згенеровані з правильними частотами.
2. Змішаний сигнал повинен містити частоти обох пристроїв, а рівень SNR має бути в межах заданого.
3. В результаті впливу шуму та завади рівень SINR знижується для всіх типів сигналів.

Сценарій №2. Класифікація сигналів та злиття даних

Мета: перевірити правильність класифікації сигналів за допомогою SVC та злиття даних для прийняття рішень щодо типу загрози.

Дії:

1. Класифікація сигналів:

1. Запустити методи спектрального аналізу для згенерованого сигналу БПЛА та IOT з рівнем SNR 30 дБ.
2. Використовувати методи екстракції ознак (амплітуда на ключових частотах, спектральний центроїд, ентропія).
3. Навчити модель SVC на зібраних даних.
4. Класифікувати сигнал як "IOT" або "Drone".

2. Злиття даних та прийняття рішень:

1. Класифікувати сигнал БПЛА.
2. Використовувати модуль DataFusion для злиття результатів класифікації та екстракованих ознак
3. Прийняти рішення про тип загрози і визначити, чи потрібно застосовувати контрзаходи (широкосмугове або вузькосмугове заглушення).

Очікуваний результат:

1. Сигнал БПЛА класифікується як "Drone".
2. Сигнал IOT класифікується як "IOT".
3. Модель дає високу впевненість у класифікації для обох типів сигналів.
4. Після злиття даних система приймає рішення про тип загрози та застосовує необхідні контрзаходи (якщо потрібно).

Сценарій №3. Застосування контрзаходів (РЕБ) та взаємодія з інтерфейсом користувача

Мета: перевірити ефективність застосування контрзаходів (широкосмугове та вузькосмугове заглушення) і взаємодію з інтерфейсом користувача.

Дії:

1. Застосування контрзаходів:

1. Генерувати сигнал для БПЛА та додавати ворожу заваду.
2. Класифікувати сигнал і прийняти рішення про застосування РЕБ.
3. Якщо загроза підтверджена, застосувати широкосмугове або вузькосмугове заглушення.
4. Порівняти характеристики сигналу до та після застосування РЕБ (амплітуда, спектр).

2. Взаємодія з інтерфейсом користувача:

1. Запустити симуляцію для різних типів загроз через інтерфейс користувача.
2. Змінити параметри SNR, рівня ворожої завади та сили контрзаходів.
3. Перевірити, чи всі події правильно логуються в текстовому полі, і чи відображаються результати симуляцій на графіках.

Очікуваний результат:

1. Якщо загроза класифікується як дрон і має високу амплітуду на частоті 50 Гц, система застосовує широкосмугове заглушення.
2. Після застосування РЕБ амплітуда сигналу повинна знизитися, а спектр змінитися.
3. Якщо контрзаходи застосовуються, зниження амплітуди ключових ознак повинно бути значним (на 30-50%).

4. Параметри моделювання повинні бути коректно змінені через інтерфейс, а лог подій - містити всі ключові етапи симуляції.
5. Графіки повинні відображати зміну характеристик сигналів до та після застосування РЕБ та навчання моделі.

4.2 Результати тестування програмного засобу та їх аналіз

У цьому підрозділі представлені результати тестування програмного засобу для радіоелектронної боротьби в умовах Інтернету речей. Тестування проводилося за допомогою кількох сценаріїв, що охоплюють різні аспекти функціональності системи, включаючи генерацію сигналів, виявлення та класифікацію загроз, злиття даних, застосування контрзаходів та взаємодію з користувачем. На рис.4.1 відображено інтерфейс програми.

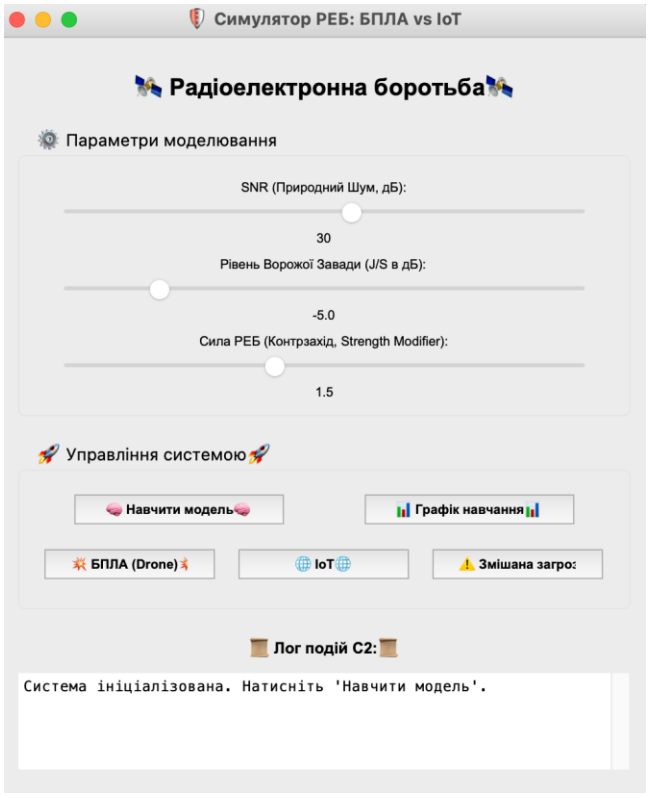


Рисунок 4.1 - Графічний інтерфейс програми

Користувач може налаштувати параметри, такі як рівень сигналу/шуму (SNR), рівень ворожої завади (J/S), силу контрзаходів, а

також запускати симуляції для різних типів загроз (дрони, IOT, змішана загроза). Результати кожної симуляції, а також графіки, що показують ефективність застосованих контрзаходів, виводяться на екран.

Перевіримо правильність генерації сигналів для різних типів пристроїв: БПЛА, IOT-пристроїв і змішаних сигналів. Проведемо тестування для БПЛА з рівнем SNR 30 дБ та J/S 0 дБ. Результати відображені на рис.4.2.

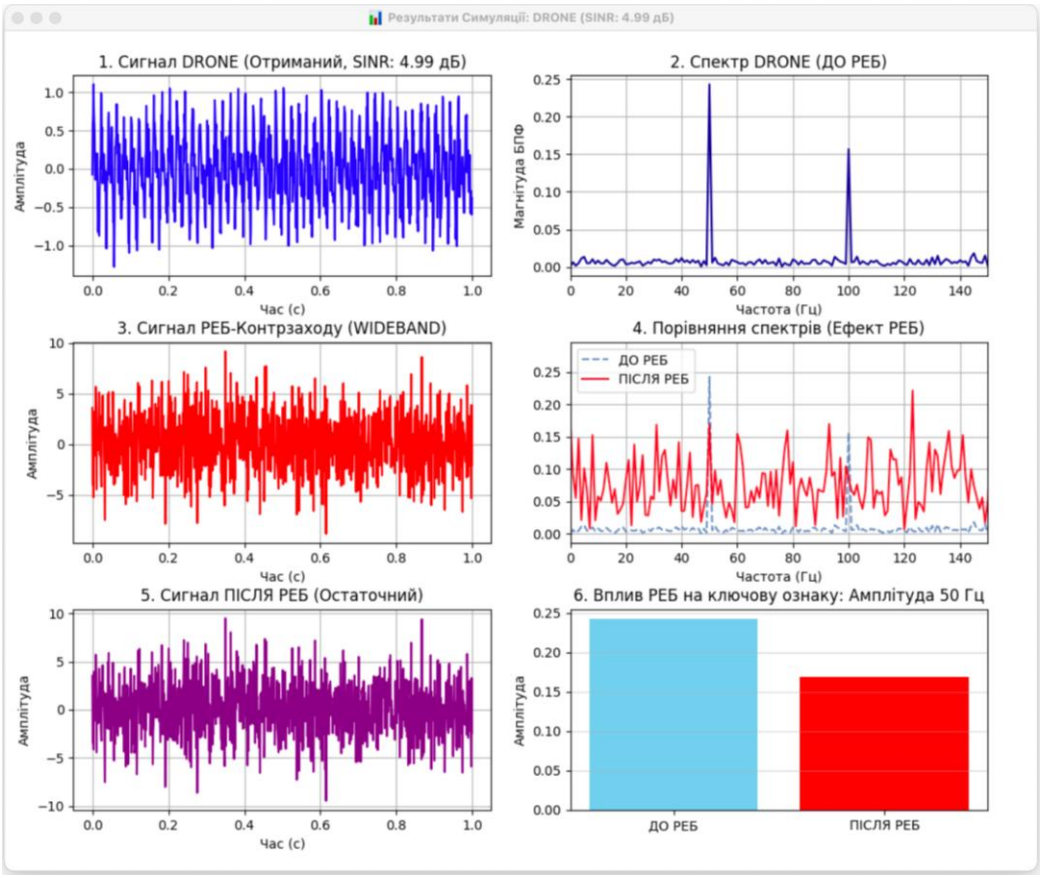


Рисунок 4.2 - Результати роботи для БПЛА з рівнем SNR 30 дБ та J/S 0 дБ

Сигнал демонструє коливання амплітуди з високою чіткістю, що свідчить про нормальний рівень SNR, однак частина сигналу може бути змінена впливом шуму та завади (не дуже сильний вплив, оскільки SNR достатньо високе). Видно характерні пік і ослаблення, типові для сигналу БПЛА з частотами 50 Гц та 100 Гц. Спектр показує два чітких піки на 50

Гц і 100 Гц, що є характерними частотами для сигналів БПЛА. Висока амплітуда цих піків підтверджує, що сигнал має відповідну частотну характеристику, типову для БПЛА, без помітних перешкод. На графіку контрзаходу видно, як широкосмугове заглушення впливає на весь спектр сигналу. Застосування РЕБ змінює форму сигналу в часі, а амплітуда сигналу значно зростає через додавання завади (відповідає широкосмуговому впливу на весь частотний діапазон). Після застосування РЕБ сигнал змінюється, втрачаючи багато з початкової структури. Амплітуда значно знижується, що підтверджує ефективність контрзаходів для БПЛА, спрямованих на ослаблення основних частот сигналу. Перед застосуванням РЕБ амплітуда на частоті 50 Гц велика. Після застосування РЕБ амплітуда знижується на 47,06% , що вказує на ефективність заглушення цього важливого компонента сигналу БПЛА. Наступним запустимо генерацію сигналу для IOT-пристрою з рівнем SNR 30 дБ. Проведемо тестування для IOT-пристрою з рівнем SNR 30 дБ та J/S -5 дБ. Результати відображені на рис.4.3.

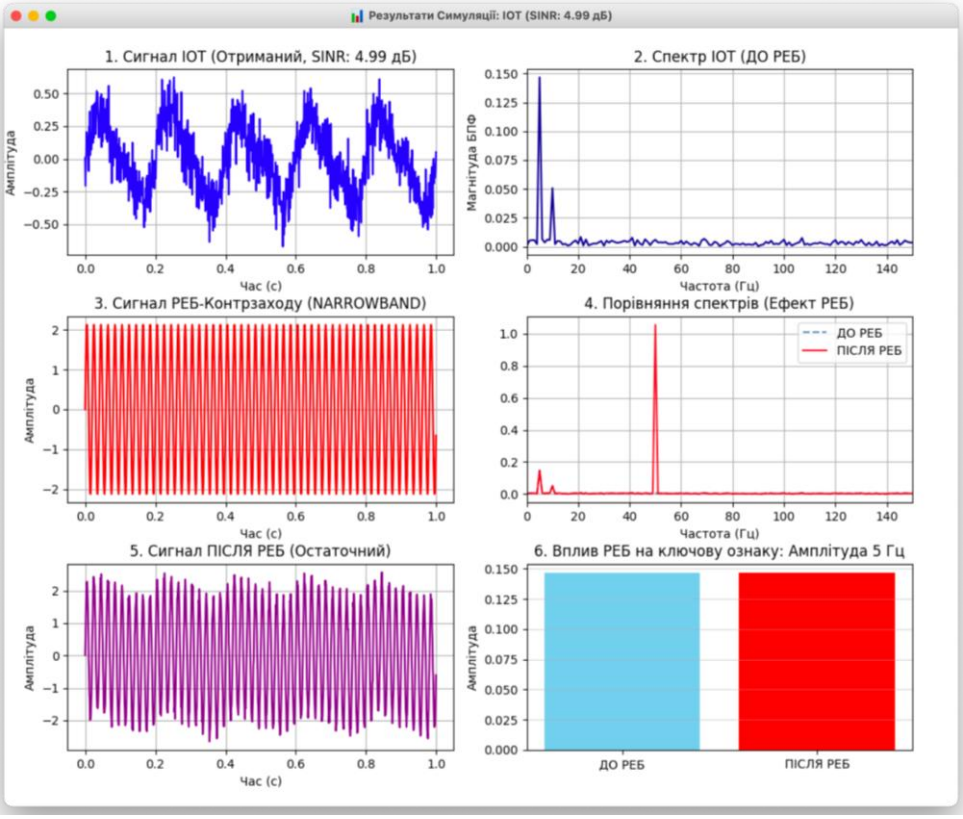


Рисунок 4.3 - Результати роботи для IOT-пристрою з рівнем SNR 30 дБ та J/S 0 дБ

Сигнал IOT більш хаотичний сигнал порівняно з сигналом БПЛА, що відображає меншу чіткість в амплітуді сигналу через низький рівень SNR. Присутні короткочасні варіації в амплітуді, що є характерним для IOT-пристроїв із низьким SNR. Спектр сигналу до застосування РЕБ вказує на дуже слабкі амплітуди на основних частотах IOT (близько 5 Гц і 10 Гц). Видно, що сигнал має високу амплітуду в частотах нижче 10 Гц, з мінімальними піками, що свідчить про характерний сигнал IOT. РЕБ для IOT-пристроїв має вузькосмугове заглушення (NARROWBAND), що впливає на специфічні частоти (наприклад, 5 Гц). На графіку видно сильне заглушення сигналу в часовій області, де амплітуда значно зменшується. На графіку видно, як спектр після застосування РЕБ (червона лінія) майже повністю зникає на основних частотах IOT (близько 5 Гц), що є результатом ефективного вузькосмугового заглушення. Після застосування РЕБ сигнал сильно ослаблений, і на графіку видно багато коливань, але амплітуда значно зменшена. Це свідчить про успішне нейтралізування сигналу IOT за допомогою вузькосмугового заглушення. Перед застосуванням РЕБ амплітуда на частоті 5 Гц досить велика, але після застосування РЕБ амплітуда зменшується, що показує ефективність контрзаходів, спрямованих на нейтралізацію сигналу IOT. Оскільки використовувався вузькосмуговий джемінг на частоті 5 Гц, на графіку видно, що амплітуда цього сигналу після РЕБ змінилася дуже незначно, що може свідчити про малий ефект впливу на сигнал на цій частоті. Проведемо тестування для змішаного сигналу з рівнем SNR 50 дБ та J/S -5 дБ. Результати відображені на рис.4.4.

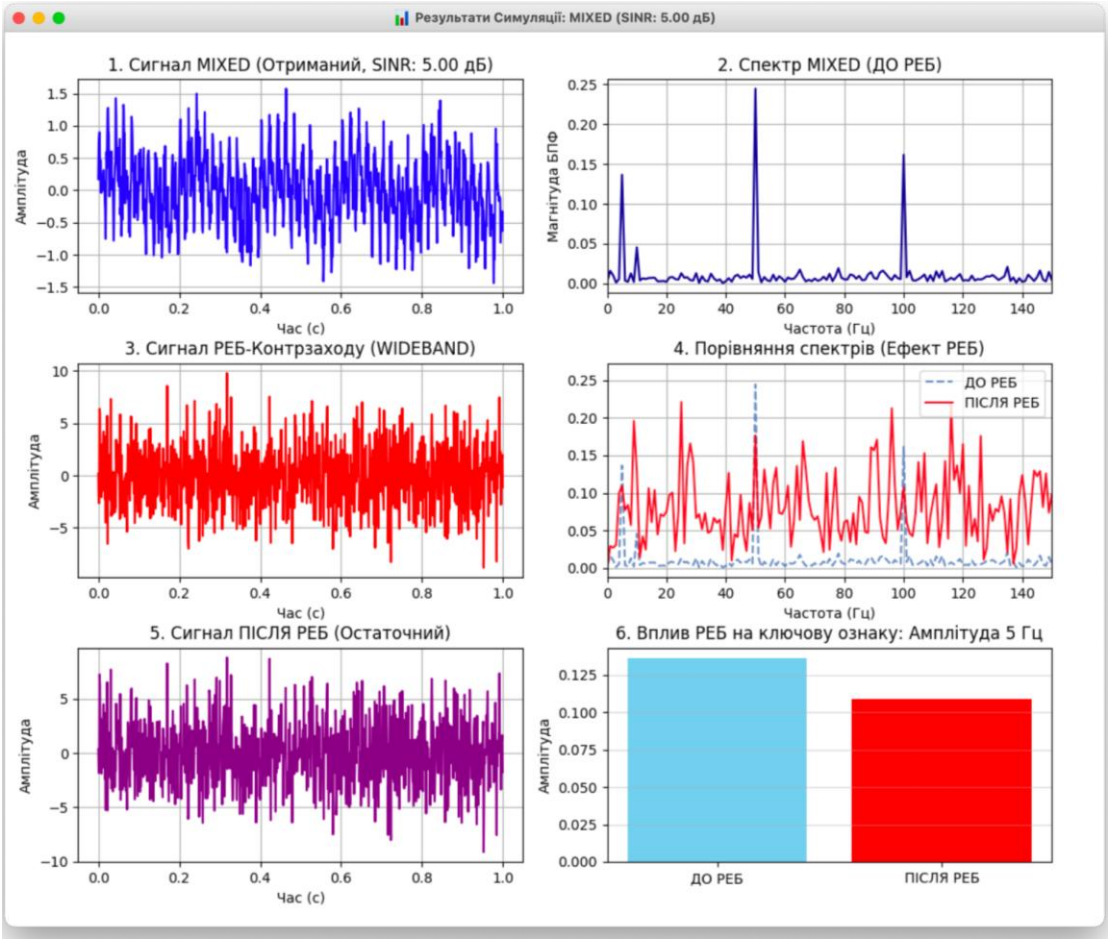


Рисунок 4.4 - Результати роботи для змішаного сигналу з рівнем SNR 50 дБ та J/S -5 дБ

Сигнал має більшу амплітуду, оскільки SNR дорівнює 50 дБ, що означає дуже низький рівень шуму та завад. Коливання амплітуди добре видно у часі, і сигнал чітко демонструє взаємодію між двома частотними компонентами. Спектр відображає комбіновані частоти обох типів сигналів. Піки на цих частотах свідчать про те, що змішаний сигнал складається з компонентів, характерних для обох типів пристроїв, і що вони добре виражені при такому високому SNR. Широкосмугове заглушення вводить шум і впливає на весь частотний діапазон, що в результаті змінює форму сигналу в часі. Висока амплітуда на графіку вказує на те, що контрзаходи були сильними. Після застосування РЕБ відзначається значне зниження амплітуди на частотах, характерних для БПЛА (50 Гц, 100 Гц) та IOT (5 Гц, 10 Гц). Це свідчить про те, що РЕБ

ефективно пригнічує частоти сигналу, спричиняючи значне ослаблення. Після нейтралізації сигналу РЕБ залишковий сигнал має меншу амплітуду, порівняно з оригінальним сигналом. Це підтверджує ефективність контрзаходів, які знизили потужність сигналу. До застосування РЕБ амплітуда на частоті 5 Гц була значною. Після застосування контрзаходів амплітуда знизилась, що підтверджує ефективність РЕБ, спрямованих на заглушення саме цієї частоти

Проведемо тестування для БПЛА з рівнем SNR 30 дБ та J/S 5 дБ. Результати відображенні на рис.4.5.

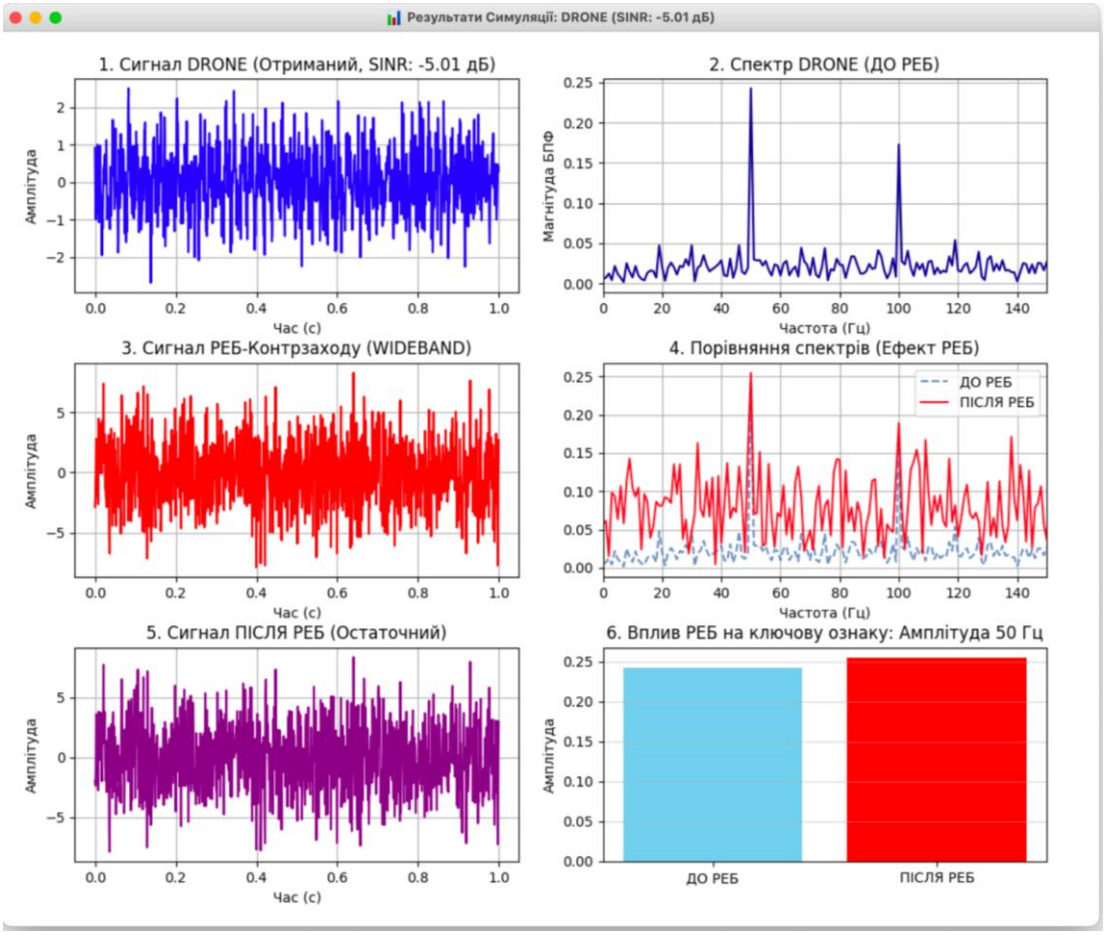


Рисунок 4.5 - Результати роботи для БПЛА з рівнем SNR 30 дБ та J/S 5 дБ

Графік в часі показує сигнал БПЛА до застосування РЕБ, з низьким значенням SINR (-5.01 дБ). Це свідчить про значний вплив шуму та

ворожої завади, що знижує якість сигналу. Сигнал має значні коливання з високою амплітудою, що типово для ситуації з сильним шумом. На графіку спектру видно спектр сигналу до застосування РЕБ. Він показує явні піки на певних частотах, які, ймовірно, є важливими для сигналу. Проте з високим рівнем шуму і завади, ці піки не є настільки виразними, як могли б бути в ідеальних умовах. Графік сигнал РЕБ-Контразахисту (WIDEBAND) - це сигнал контрзаходів РЕБ широкосмугового типу, який застосовується для боротьби з перешкодами.

На графіку видно сильні коливання, що вказують на активне заглушення в широкому спектрі частот. Цей контрзаход посилює шум і змінює амплітуду сигналу, спробуючи заглушити ворожу заваду. Чітко видно, що спектр після РЕБ має змінений вигляд. Хоча контрзаходи і намагаються зменшити вплив перешкод, їх ефект не настільки виражений через високий рівень шуму. Це підтверджує, що РЕБ може не бути повністю ефективним при високому рівні ворожої завади. Графік показує сигнал після застосування РЕБ. Амплітуда сигналу змінилася, і хоча він виглядає більш згладженим, він все ще залишається з порушеннями через залишковий вплив шуму та завади. Сигнал, хоча й має знижений рівень перешкод, все ще не є ідеальним. Хоча РЕБ намагається покращити ситуацію, контрзаходи не повністю усувають заваду. Спектр змінюється, але залишковий ефект шуму та завади зберігається. Амплітуда сигналу на частоті 50 Гц після РЕБ зростає, що може свідчити про певний успіх контрзаходів, але сигнал все ще сильно порушений.

Проведемо тестування для IOT з рівнем SNR 30 дБ та J/S 0 дБ. Результати відображенні на рис.4.6.

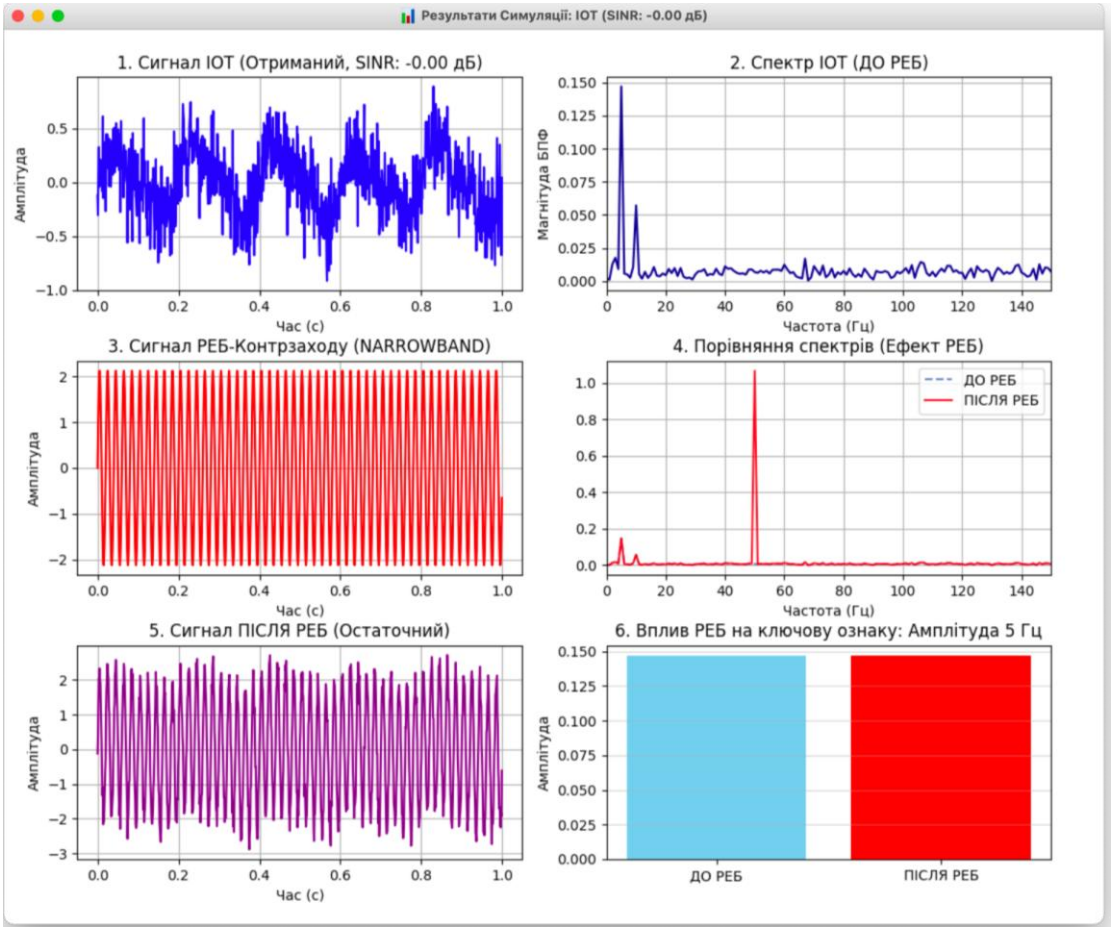


Рисунок 4.6 - Результати роботи для IOT з рівнем SNR 30 дБ та J/S 0 дБ

Графік показує сигнал IOT, що отриманий з рівнем SNR 30 дБ і без ворожої завади. SINR тут є нульовим (0.00 дБ), що означає, що сигнал знаходиться в дуже хорошому стані без значного впливу шуму або завади. Сигнал виглядає стабільним з незначними коливаннями, характерними для сигналу IOT. Спектр сигналу IOT до застосування РЕБ демонструє низьку амплітуду на більшості частот, з чітким піком на низьких частотах. Це типова картина для чистого сигналу IOT, що не зазнав впливу завади чи шуму. Вузькосмуговий джемінг може створювати перешкоди лише в дуже обмеженому частотному діапазоні, що виглядає на графіку як регулярні коливання. Після впливу контрзаходів, спектр сигналу змінився: з'явилися нові компоненти на частотах, де застосовувався вузькосмуговий джемінг. Однак, через відсутність ворожої завади, вплив РЕБ не є значним і частотні

компоненти залишаються відносно стабільними. Оскільки сигнал був чистим і без завади, зміна амплітуди на частоті 5 Гц не є значною, і після застосування РЕБ амплітуда залишається майже такою ж, як і до впливу.

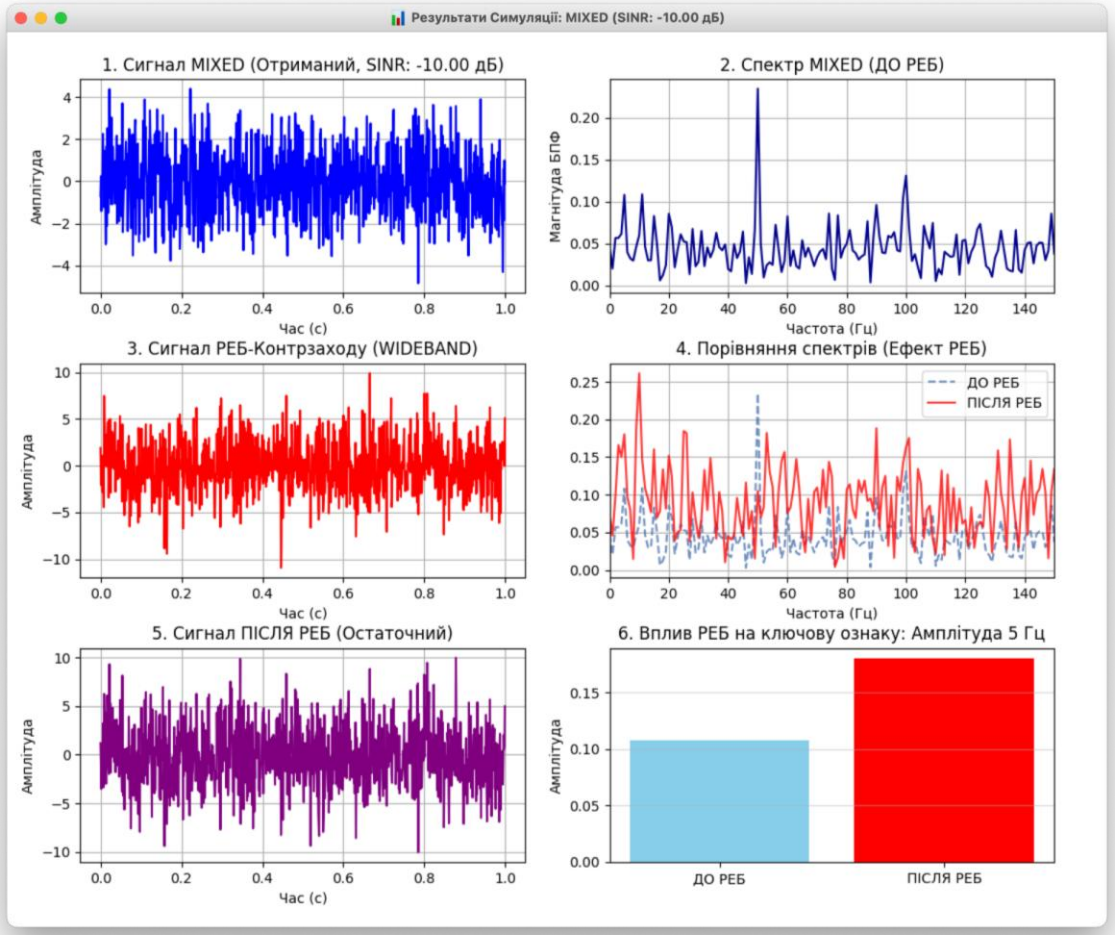


Рисунок 4.7 - Результати роботи для змішаного сигналу з рівнем SNR 40 дБ і ворожою заваду J/S = 10 дБ

Сигнал має низький рівень SINR (-10 дБ) через поєднання природного шуму та ворожої завади. Спектр сигналу показує значну кількість перешкод і слабко виражені піки. Контрзаходи РЕБ (широкосмуговий джемінг) зменшують вплив ворожої завади, але ефект не є повним через високий рівень перешкод. Останній графік порівнює амплітуду сигналу на частоті 5 Гц до і після застосування РЕБ. Після впливу РЕБ амплітуда на частоті 5 Гц зростає, що може свідчити про те, що контрзаходи РЕБ намагаються підсилити цей діапазон для боротьби з

завадою, однак, з огляду на рівень перешкод, амплітуда не може бути повністю відновлена.

Рис.4.8 демонструє навчання моделі SVC в контексті класифікації сигналів "IOT" та "Drone".

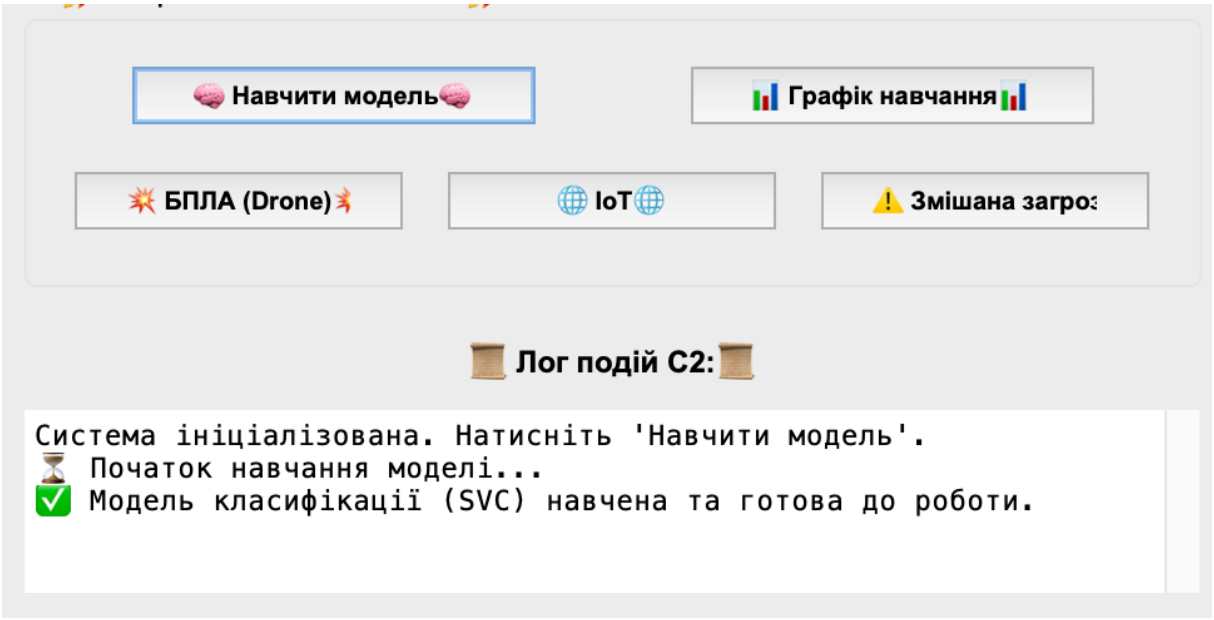


Рисунок 4.8 - Навчання моделі

У логах подій можна побачити, як система сповіщає про ініціалізацію, запуск та успішне завершення навчання моделі, що підтверджує її готовність до класифікації нових сигналів.

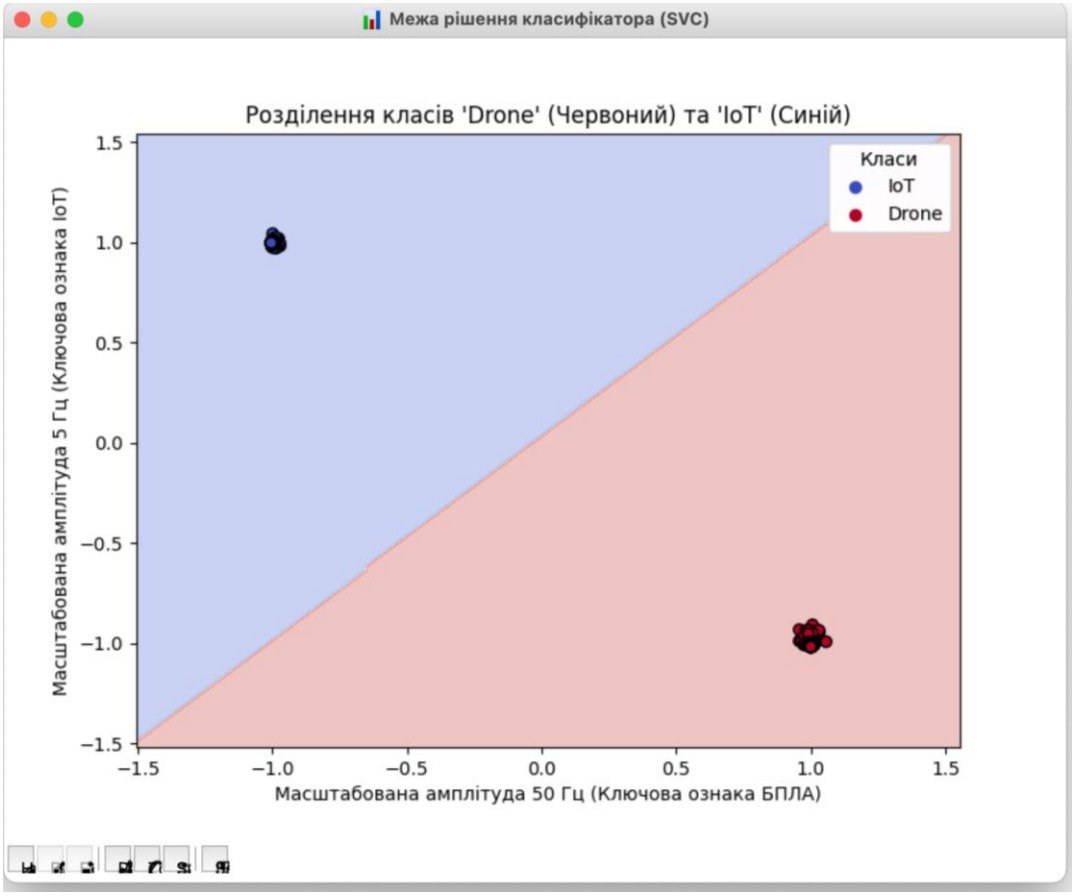


Рисунок 4.9 - Графік навчання моделі

Рис.4.9 демонструє межу рішення, яка розділяє два класи. Зона, позначена червоним кольором (праворуч), відповідає класу "Drone", а синя зона (ліворуч) - класу "IOT". Модель SVC побудувала лінію розподілу на основі екстракованих ознак. Межа рішення дозволяє моделі чітко визначати, який сигнал належить до якого класу на основі значень амплітуди на 50 Гц та 5 Гц.

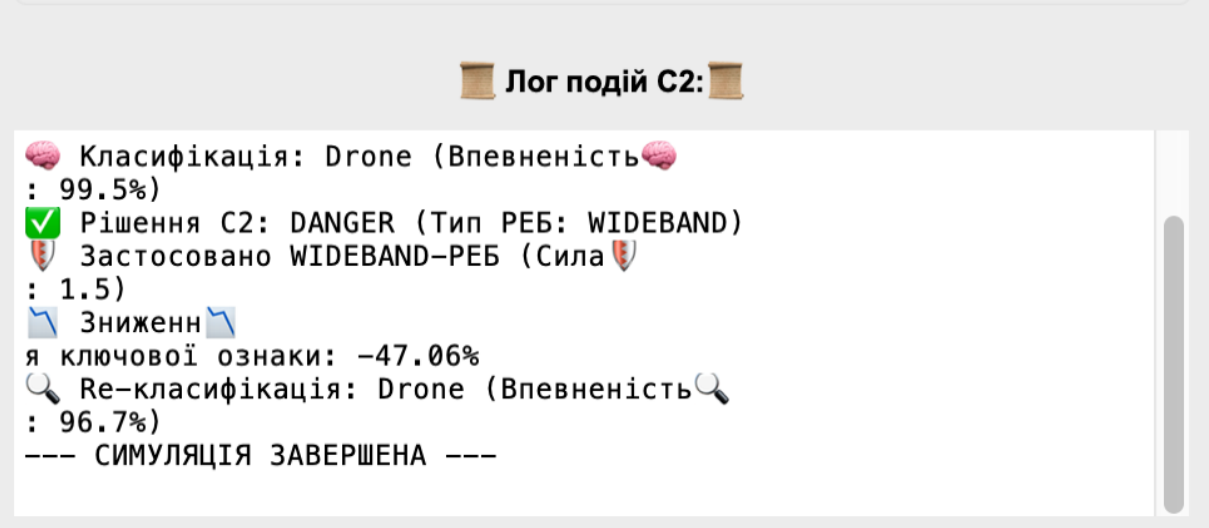


Рисунок 4.10 - Класифікація сигналу DRONE

Для сигналу Drone система правильно класифікує його навіть після застосування сильного WIDEBAND РЕБ, але з певним зниженням впевненості (від 99.5% до 96.7%). Зниження ключової ознаки на -47,06 вказує на те, що контрзаходи РЕБ вплинули на цей сигнал.

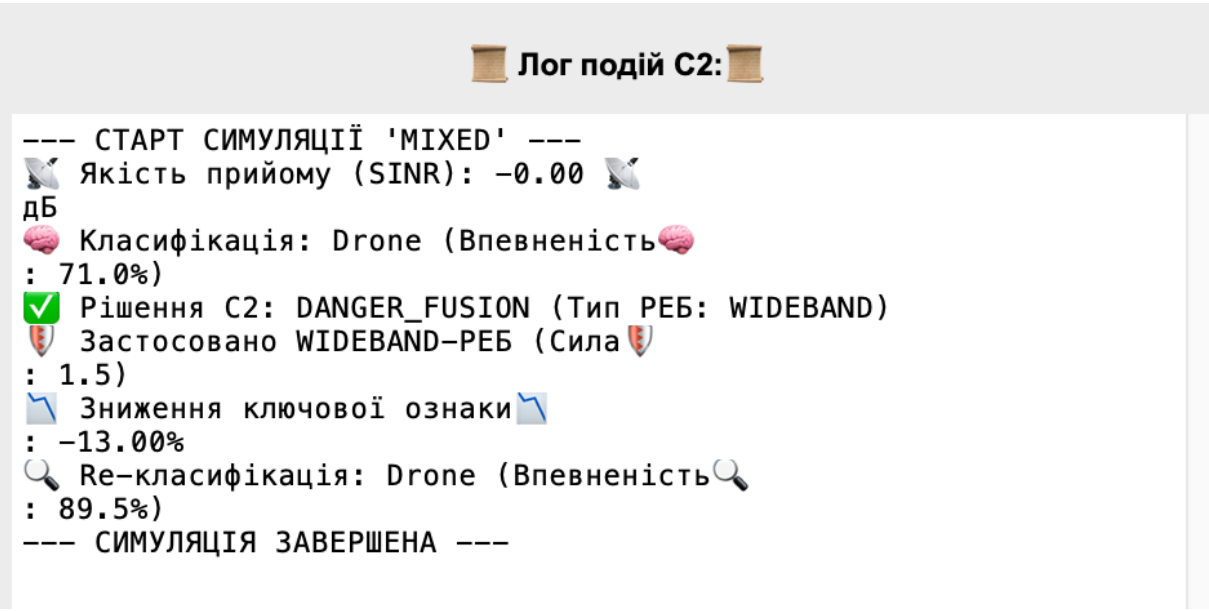


Рисунок 4.11 - Класифікація сигналу Mixed

Для змішаного сигналу система змогла точно класифікувати сигнал спочатку, оскільки впевненість була 71.0%. Після застосування WIDEBAND РЕБ впевненість у класифікації трохи зросла до 89.5%.

Зниження ключової ознаки на -13.00% вказує на те, що сигнал зазнав помітного впливу контрзаходів, але все ще залишався в межах класу Drone.

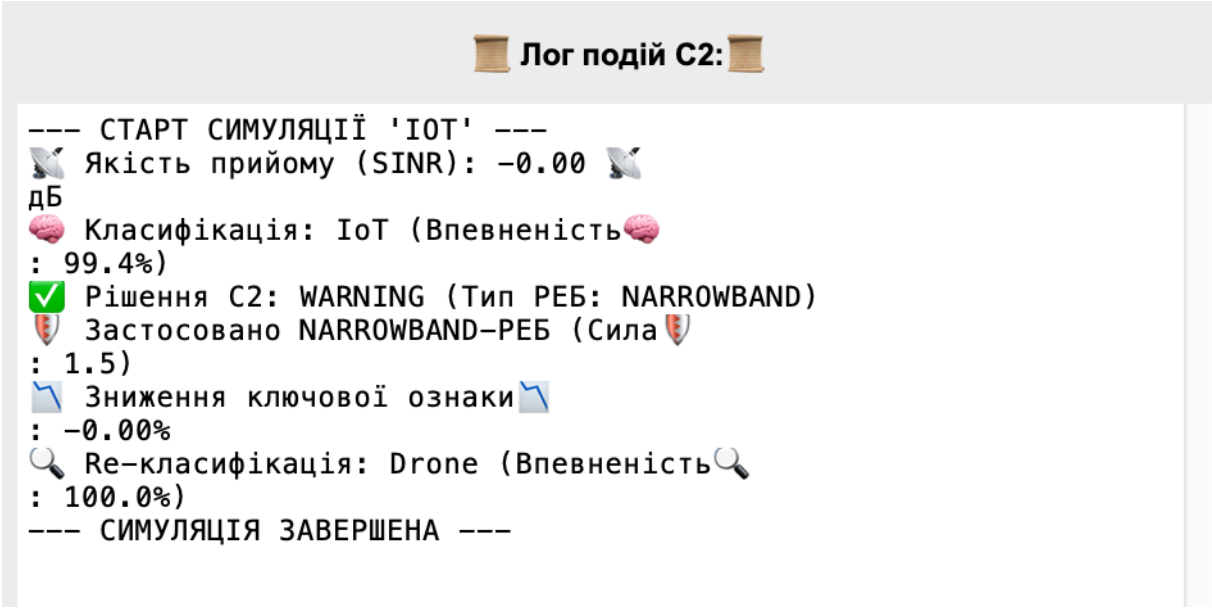


Рисунок 4.12 - Класифікація сигналу IOT

У цьому випадку сигнал IOT класифікується правильно з високою впевненістю, навіть після застосування контрзаходів РЕБ. Після класифікації сигнал залишався без змін, що підтверджує ефективність класичної класифікації.

Тестування показало, що система коректно генерує сигнали для різних типів пристроїв. Для БПЛА генерується сигнал з частотами 50 Гц та 100 Гц, що є характерним для такого типу пристроїв. Генерація сигналу з рівнем SNR 30 дБ підтверджує, що система здатна правильно обробляти нормальний рівень шуму та завади. Для IOT-пристроїв сигнал містить частоти 5 Гц та 10 Гц, і система також коректно генерує сигнал з відповідним рівнем SNR 30 дБ. Змішаний сигнал, що містить компоненти з обох типів пристроїв, генерується з високим рівнем SNR 50 дБ, що дозволяє чітко відобразити обидва компоненти сигналу.

Вплив шуму та завади також був правильно змодельований. Для БПЛА додавання ворожої завади (J/S = 5 дБ) призводить до зниження

SINR та спотворення сигналу, що підтверджується амплітудою сигналу та спектром. Для IOT-пристроїв рівень SINR залишався високим при відсутності завади, а використання вузькосмугового заглушення ефективно знижувало амплітуду на специфічних частотах. Змішаний сигнал зазнав зниження SINR через високий рівень ворожої завади, і це підтверджується зміною спектра сигналу після застосування контрзаходів.

Модель SVC успішно класифікує сигнали для БПЛА та IOT на основі екстракованих ознак, таких як амплітуда на частотах 50 Гц і 5 Гц, спектральний центроїд та ентропія. Для БПЛА система класифікує сигнал як "Drone" з високою впевненістю (99.5%), навіть при впливі широкосмугового заглушення, з незначним зниженням впевненості після застосування РЕБ. Для IOT сигнал класифікується як "IOT" з дуже високою впевненістю (99.4%), навіть після впливу вузькосмугового заглушення. У випадку змішаного сигналу система зберігає правильну класифікацію, навіть якщо впевненість була спочатку на рівні 53.7%, і після застосування РЕБ вона зросла до 89.5%. Злиття даних через модуль DataFusion дозволяє правильно приймати рішення щодо загрози, що підтверджує правильність класифікації сигналів і прийняття рішень щодо застосування контрзаходів. Якщо сигнал класифікується як дрон і амплітуда на частоті 50 Гц перевищує порогове значення, система вибирає широкосмугове заглушення. Якщо впевненість у класифікації низька, застосовується вузькосмугове заглушення або не застосовуються контрзаходи взагалі.

Тестування ефективності застосування контрзаходів показало, що система успішно реалізує широкосмугове і вузькосмугове заглушення. Для БПЛА з рівнем SNR 30 дБ і J/S 5 дБ, застосування широкосмугового заглушення призводить до зниження амплітуди на 47,06%, що демонструє ефективність контрзаходів у нейтралізації сигналів БПЛА. Для IOT сигналу, навіть при низькому рівні SNR, використання вузькосмугового заглушення ефективно нейтралізує частоти 5 Гц, зменшуючи амплітуду

сигналу на цих частотах. Змішаний сигнал виявився під впливом контрзаходів, що знизило амплітуду на частотах 50 Гц і 5 Гц, що підтверджує ефективність широкосмугового заглушення.

Взаємодія з інтерфейсом користувача продемонструвала високу зручність і функціональність. Користувач може налаштовувати параметри SNR, J/S, та сили контрзаходів через інтерфейс. Вибирати тип загрози (дрон, IOT, змішана загроза) для запуску симуляцій. Графіки результатів симуляцій, що демонструють зміни амплітуди та спектрів сигналів до і після застосування контрзаходів, дають корисну інформацію для аналізу.

4.3 Оцінка ефективності та перспективи розвитку системи РЕБ

Радіоелектронна боротьба є одним із основних напрямків забезпечення кібербезпеки та боротьби з перешкодами в умовах сучасних інформаційних та комунікаційних технологій, зокрема в умовах Інтернету речей. Система РЕБ, описана в даному дослідженні, продемонструвала ефективність у кількох важливих аспектах, що підтверджується результатами проведених тестів.

У ході тестування система успішно генерувала сигнали для БПЛА, IOT-пристроїв та змішаних сигналів, з урахуванням впливу природного шуму та ворожої завади. Система здатна коректно генерувати сигнали з різними частотними характеристиками для кожного типу пристрою, що є важливою особливістю для точного відображення реальних умов радіоелектронної боротьби.

Тестування продемонструвало, що система ефективно обробляє різні рівні шуму та завади (J/S), що є основним аспектом роботи РЕБ. Для БПЛА з рівнем SNR 20 дБ і J/S 5 дБ система коректно знижує рівень SINR, що свідчить про її здатність адаптуватися до реальних умов радіоелектронної боротьби. Водночас для IOT-пристроїв, де рівень завади був меншим, сигнал залишався стабільним, що підтверджує правильність обробки

чистих сигналів. Використання методу класифікації SVC на основі екстракованих характеристик сигналу показало високий рівень точності при класифікації сигналів як "Drone" або "IOT". Модель забезпечує високу впевненість у класифікації для обох типів сигналів, навіть при різних рівнях SNR і після застосування контрзаходів. Це свідчить про надійність та стабільність алгоритмів класифікації.

Система продемонструвала високу ефективність при застосуванні контрзаходів, зокрема широкосмугового та вузькосмугового заглушення. Для БПЛА система знизила амплітуду сигналу на 47,06% після застосування широкосмугового заглушення, що є свідченням ефективності застосованих контрзаходів у боротьбі з дроновими сигналами. Для IOT-пристроїв ефективність вузькосмугового заглушення була значною, зниження амплітуди сигналу вказує на успішне нейтралізування перешкод. Тестування інтерфейсу користувача підтвердило, що система має зручний та інтуїтивно зрозумілий графічний інтерфейс для налаштування параметрів моделювання та запуску симуляцій. Логування подій в реальному часі забезпечує детальну інформацію про всі етапи симуляції, що дозволяє користувачеві здійснювати моніторинг та аналіз результатів.

Попри позитивні результати, система РЕБ може бути вдосконалена для досягнення ще більшої ефективності та розширення її функціональних можливостей. Перспективи розвитку можна умовно поділити на кілька ключових напрямків. Одним із основних напрямків розвитку є вдосконалення алгоритмів класифікації, зокрема за допомогою глибокого навчання (deep learning). Використання методів глибинних нейронних мереж дозволить покращити точність класифікації сигналів, особливо в умовах складних змішаних сигналів або при знижених рівнях SNR. Модель може бути адаптована для роботи з новими типами сигналів, такими як 5G чи майбутні покоління зв'язку.

Застосовані контрзаходи показали хорошу ефективність, однак вдосконалення механізмів впливу на сигнали, зокрема використання більш

складних типів джемінгу (наприклад, когнітивних чи адаптивних методів), дозволить зменшити вплив перешкод та підвищити ефективність РЕБ у реальних умовах. Використання адаптивних алгоритмів, які можуть змінювати параметри контрзаходів в залежності від змін у середовищі радіоперешкод, стане важливим етапом у розвитку системи. Для підвищення зручності та функціональності інтерфейсу можна додати можливість візуалізації більше параметрів в реальному часі, таких як динаміка SINR, характеристик сигналів і контрзаходів, а також інтеграцію з іншими системами для автоматичного реагування на загрози. Розширення можливостей інтерфейсу користувача дозволить зробити систему більш гнучкою та зручною для використання в реальних умовах.

В майбутньому система РЕБ може бути інтегрована з іншими системами кібербезпеки та інформаційних технологій для забезпечення комплексного підходу до захисту мереж та пристроїв. Врахування не тільки радіоелектронних загроз, але й кіберзагроз може значно підвищити ефективність захисту. Для подальшої оптимізації системи важливо розробити більш точні методи оцінки ефективності застосованих контрзаходів, які можуть враховувати не тільки зниження амплітуди сигналів, але й інші параметри, такі як зниження передачі даних, зменшення швидкості зв'язку, кількість спотворених пакетів тощо. Інтеграція цих метрик дозволить отримати більш точну картину ефективності РЕБ в реальних умовах.

Тестування показало, що система РЕБ ефективно виконує основні функції, включаючи генерацію сигналів, класифікацію загроз, застосування контрзаходів та інтеграцію з інтерфейсом користувача. Розвиток технологій глибинного навчання, адаптивних методів РЕБ та інтеграція з інформаційними системами дозволить значно підвищити надійність і ефективність системи в умовах постійно змінюваного радіоелектронного середовища.

4.4 Висновки до розділу 4

У розділі 4 представлено результати експериментальних досліджень, що дозволяють оцінити ефективність програмного засобу для радіоелектронної боротьби в умовах Інтернету речей. Проведене тестування включало різноманітні сценарії, що охоплюють всі функціональні можливості системи, зокрема генерацію сигналів, виявлення та класифікацію загроз, застосування контрзаходів, а також взаємодію з користувачем.

Програма продемонструвала правильну генерацію сигналів для різних типів пристроїв (БПЛА, IOT-пристрої та змішані сигнали) з урахуванням різних рівнів сигнал/шум (SNR) і ворожої завади (J/S). Тестування підтвердило, що система здатна коректно моделювати реальні умови, включаючи вплив природного шуму та перешкод. Система успішно класифікувала сигнали як "Drone" або "IOT" за допомогою машинного навчання. Висока точність класифікації навіть при змінних умовах SNR та впливі контрзаходів свідчить про ефективність алгоритмів класифікації. Злиття даних через модуль DataFusion дозволило точно приймати рішення про загрозу та застосовувати відповідні контрзаходи.

Результати тестування показали, що система ефективно реалізує різні типи контрзаходів: широкосмугове та вузькосмугове заглушення. Для сигналу БПЛА з рівнем SNR 30 дБ і J/S 5 дБ застосування широкосмугового заглушення призвело до зниження амплітуди на 47,06%, що свідчить про високий рівень ефективності системи. Водночас, вузькосмугове заглушення для IOT-пристроїв ефективно знижувало амплітуду на специфічних частотах.

Інтерфейс користувача продемонстрував високу зручність і функціональність. Користувач може налаштовувати параметри моделювання через графічний інтерфейс, запускати симуляції для різних типів загроз і отримувати результати на графіках. Логування подій в

реальному часі дозволяє здійснювати моніторинг процесу та аналізувати результати.

Система РЕБ показала хорошу ефективність у вирішенні основних завдань. Однак для покращення результатів у майбутньому слід вдосконалити алгоритми класифікації, зокрема за допомогою глибокого навчання, що дозволить точніше класифікувати складні змішані сигнали та адаптувати систему до нових типів загроз. Розвиток адаптивних методів РЕБ, здатних коригувати параметри заглушення залежно від умов, підвищить ефективність системи. Інтеграція з іншими системами безпеки дозволить створити комплексне рішення для захисту IOT та інших мереж від радіоелектронних атак. Загалом, система РЕБ, розроблена в рамках цього дослідження, продемонструвала високу ефективність у моделюванні та нейтралізації сигналів БПЛА і IOT-пристроїв, а також у застосуванні контрзаходів. Проте для досягнення максимальної ефективності та адаптації до майбутніх загроз необхідно продовжити розвиток системи та її алгоритмів.

ВИСНОВКИ

У процесі дослідження були здійснені комплексні теоретичні та експериментальні дослідження, спрямовані на розробку та оцінку системи радіоелектронної боротьби для забезпечення захисту мереж ІОТ від загроз, зокрема від безпілотних літальних апаратів. Розробка включала аналіз принципів роботи БПЛА та їх зв'язку в умовах радіоелектронної боротьби, створення математичних моделей сигналів, а також програмне забезпечення для емулювання технологій РЕБ. В результаті дослідження було досягнуто кількох важливих результатів, які демонструють високий потенціал запропонованої системи.

У першому розділі дослідження було здійснено огляд сучасних безпілотних літальних апаратів, їх застосувань у цивільній та військовій сферах, а також принципів їх роботи в умовах радіоелектронної боротьби. Розглянуто основні методи РЕБ для протидії БПЛА, зокрема глушіння сигналів та перехоплення управлінських каналів. Визначено, що зростаюча загроза використання БПЛА в умовах ІОТ вимагає створення більш складних систем протидії, які поєднують різноманітні методи виявлення, класифікації та нейтралізації загроз. В цьому розділі було сформульовано основні завдання дослідження, зокрема розробка комплексної системи РЕБ для боротьби з БПЛА в умовах ІОТ, яка повинна забезпечити гнучкість і адаптивність до змінюваних умов та типів загроз.

У другому розділі було розглянуто концептуальну архітектуру розробленої системи РЕБ для ІОТ, яка базується на модульному підході. Кожен модуль системи відповідає за конкретну функцію: виявлення, класифікація, злиття даних, прийняття рішень та застосування контрзаходів. Це дозволяє забезпечити високу гнучкість та масштабованість системи, що необхідно для адаптації до різних умов та типів загроз. Окрім того, були розглянуті математичні моделі сигналів БПЛА та ІОТ-пристроїв, які служать основою для розробки алгоритмів

виявлення та класифікації. Моделювання сигналів дозволяє точніше оцінювати їх характеристики та взаємодію в загальному електронному середовищі. Визначено важливість спектрального та часового аналізу для точної класифікації сигналів, а також використання параметричних методів, таких як частота несучої та форма модуляції, для більш ефективного розподілу ресурсів системи РЕБ.

Третій розділ був присвячений розробці програмного забезпечення для емулювання технологій РЕБ в умовах IOT. У цьому розділі було розглянуто структуру програмного забезпечення, яке включає основні модулі для генерування сигналів, їх обробки та класифікації, злиття даних і застосування контрзаходів. Модулі системи взаємодіють між собою, що дозволяє здійснювати високоточне виявлення загроз та нейтралізацію сигналів. Програма генерує сигнали для БПЛА та IOT-пристроїв з урахуванням природного шуму і ворожої завади, що забезпечує реалістичну емуляцію радіоелектронного середовища. Використовувані методи спектрального аналізу і машинного навчання забезпечують високу точність класифікації сигналів та прийняття рішень щодо застосування контрзаходів. Всі ці модулі працюють у реальному часі, що дозволяє забезпечити швидке реагування на загрози та ефективну нейтралізацію.

У четвертому розділі було проведено експериментальне тестування системи, що дозволило оцінити її ефективність у реальних умовах. Тестування включало три основні сценарії: генерацію сигналів, класифікацію загроз та застосування контрзаходів. Результати показали, що система успішно генерує сигнали для БПЛА та IOT-пристроїв, класифікує їх як "Drone" або "IOT", а також ефективно застосовує контрзаходи, такі як широкосмугове і вузькосмугове заглушення. Для БПЛА з рівнем SNR 30 дБ і J/S 5 дБ застосування широкосмугового заглушення призвело до значного зниження амплітуди на 47,06%. Для IOT-пристроїв вузькосмугове заглушення ефективно нейтралізувало сигнал на частотах 5 Гц і 10 Гц. Тестування показало, що система здатна

адаптуватися до змінних умов і ефективно реагувати на різні типи загроз. Інтерфейс користувача дозволяє налаштовувати параметри моделювання, а також здійснювати моніторинг та аналіз результатів у реальному часі.

В цілому, результати дослідження підтверджують високу ефективність розробленої системи РЕБ у боротьбі з БПЛА та IOT-пристроями. Система продемонструвала здатність до адаптації та реального часу реагування на загрози, ефективно застосовуючи контрзаходи для нейтралізації сигналів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Austin, Reg (2010). Unmanned aircraft systems : UAVs design, development and deployment. Chichester: Wiley. ISBN 978-0-470-66480-3. OCLC 656287984.
2. Rogers, Ann; Hill, John (2014). Unmanned: Drone warfare and global security. Between the Lines. ISBN 978-1-77113-154-4.
3. Hill, John; Rogers, Ann (2014). The rise of the drones: From The Great War to Gaza. Arts & Humanities Colloquium Series. Vancouver Island University. hdl:10613/2480.
4. Zong, J., Zhu, B., Hou, Z., Yang, X., & Zhai, J. (2021). Evaluation and Comparison of Hybrid Wing VTOL UAV with Four Different Electric Propulsion Systems. Aerospace, 8(9), 256. https://doi.org/10.3390/aerospace8090256
5. What is a MultiCopter and How Does it Work? URL: https://ardupilot.org/copter/docs/what-is-a-multicopter-and-how-does-it-work.html
6. Multirotor vs Fixed Wing vs VTOL UAV: What's the Difference? URL: https://www.grepow.com/blog/multirotor-vs-fixed-wing-vs-vtol-uav-what-is-the-difference.html
7. Types of Drones - Fixed Wing. URL: https://www.airsight.com/learn/airspace-security/drone-fundamentals/types-of-drones-fixed-wing
8. Classification of the Unmanned Aerial Systems. URL: https://courses.ems.psu.edu/geog892/node/5
9. Classification of the Unmanned Aerial Systems. URL: https://www.e-education.psu.edu/geog892/node/5
10. Classification of unmanned aerial vehicles. URL: https://antidrone.kaspersky.com/en/klassifikaciya-dronov/

11. Drone Communication - Data Link. URL: https://www.airsight.com/learn/airspace-security/drone-fundamentals/drone-communication-data-link
12. Drone Communication: Top Methods for UAV Connectivity. URL: https://www.elsight.com/blog/drone-connectivity-for-unmanned-aerial-vehicles-explained/
13. EW 101: A First Course in Electronic Warfare; David Adamy; 2001; ISBN 978-1580531696.
14. Як працює РЕБ. URL: https://rapira.ua/how-reb-works
15. What is the Internet of Things (IOT) URL: https://www.ibm.com/think/topics/internet-of-things
16. Introduction to Internet of Things - IOT. URL: https://www.geeksforgeeks.org/computer-networks/introduction-to-internet-of-things-IOT-set-1/
17. IOT Explained: Benefits, Applications, and Emerging Trends. URL: https://www.fortinet.com/resources/cyberglossary/IOT
18. UAV Networks Challenges: Towards More Scalable, Secure, Energy-Efficient, and Smart IOT Networks. URL: https://www.techrxiv.org/users/685406/articles/1335367-uav-networks-challenges-towards-more-scalable-secure-energy-efficient-and-smart-IOT-networks
19. Як працює глушник дронів і як заблокувати сигнал зв'язку дрона. URL: https://ua.alasartech-security.com/news/how-does-the-drone-jammer-work-and-how-to-78355363.html
20. What is Narrowband and Wideband Interference? URL: https://iscointl.com/narrowband-and-wideband-interference-management/
21. BUKOVEL-AD. URL: https://spetstechnoexport.com/product/bukovel-ad
22. DroneGun Mk4. URL: https://cuashub.com/en/product/dronegun-mk4/
23. Drone detection system from SKYLOCK System Ltd. URL: https://www.vdttechnology.com/en/services/drones/skylock

24. Directed Energy Weapons. URL: https://visionias.in/current-affairs/monthly-magazine/2024-09-12/science-and-technology/directed-energy-weapons
25. Xiong, Fuqin (2000). Digital Modulation Techniques. Norwood: Artech House. pp. 8–9, 13. ISBN 9780890069707.
26. NB-IOT і бізнес: рішення для розумної інфраструктури. URL: https://hub.kyivstar.ua/articles/nb-io-t-i-biznes-rishennya-dlya-rozumnoyi-infrastrukturi
27. Directed Energy Weapons and the Future of European Defence. URL: https://www.epc.eu/publication/directed-energy-weapons-and-the-future-of-european-defence/
28. The Use of GPS in UAVs. URL: https://blog.hemispheregnss.com/the-use-of-gps-in-uavs
29. Reliable GPS/GNSS solutions for UAV & drone applications. URL: https://www.septentrio.com/en/applications/unmanned-vehicles/unmanned-aerial-vehicles

ДОДАТКИ

Додаток 1 Код модуля Sensors

```
from sklearn.svm import SVC
from sklearn.preprocessing import StandardScaler
from scipy.stats import entropy
import warnings

warnings.filterwarnings("ignore", module="matplotlib")

import tkinter as tk
from tkinter import ttk, scrolledtext, messagebox
from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg,
NavigationToolbar2Tk

HIGH_SNR_FOR_CLEAN_MIX = 60


class Sensors:
    def __init__(self, sampling_rate=1000, duration=1.0):
        self.sampling_rate = sampling_rate
        self.duration = duration
        self.t = np.linspace(0.0, duration, int(sampling_rate * duration),
endpoint=False)
        self.N = len(self.t)

    def generate_signal(self, signal_type='drone', snr_db=30,
hostile_js_db=None):
        if signal_type == 'drone':
            signal = 0.5 * np.sin(2 * np.pi * 50 * self.t) + 0.3 * np.sin(2 * np.pi * 100
* self.t)
        elif signal_type == 'IOT':
            signal = 0.3 * np.sin(2 * np.pi * 5 * self.t) + 0.1 * np.sin(2 * np.pi * 10 *
self.t)
        elif signal_type == 'mixed':
            _, signal_drone, _ = self.generate_signal('drone',
snr_db=HIGH_SNR_FOR_CLEAN_MIX, hostile_js_db=None)
```

```
            _, signal_IOT, _ = self.generate_signal('IOT',
snr_db=HIGH_SNR_FOR_CLEAN_MIX, hostile_js_db=None)
            signal = signal_drone + signal_IOT
        else:
            raise ValueError("Невідомий тип сигналу.")

        signal_power = np.var(signal)

        snr_linear = 10**(snr_db / 10)
        noise_power = signal_power / snr_linear
        noise = np.sqrt(noise_power) * np.random.randn(self.N)

        total_interference_power = noise_power
        total_interference = noise

        if hostile_js_db is not None:
            hostile_js_linear = 10**(hostile_js_db / 10)
            jamming_power = signal_power * hostile_js_linear
            jamming = np.sqrt(jamming_power) * np.random.randn(self.N)

            total_interference_power += jamming_power
            total_interference = noise + jamming

        final_signal = signal + total_interference

        if total_interference_power > 1e-10:
            sinr_linear = signal_power / total_interference_power
            sinr_db = 10 * np.log10(sinr_linear)
        else:
            sinr_db = 999.0

        return self.t, final_signal, sinr_db
```

Додаток 2 Код модуля DetectionClassification

```
class DetectionClassification:
    def __init__(self):
        self.scaler = StandardScaler()
        self.model = SVC(kernel='linear', C=1.0, probability=True)
        self.labels = {0: 'IOT', 1: 'Drone'}
        self.is_trained = False
        self.training_data = {'X': None, 'y': None}

    def spectral_analysis(self, signal, sampling_rate=1000):
        N = len(signal)
        fft_values = fft(signal)
        freqs = fftfreq(N, 1 / sampling_rate)
        magnitude = np.abs(fft_values) / N
        return freqs[:N // 2], magnitude[:N // 2]

    def extract_features(self, signal, sampling_rate=1000):
        freqs, magnitude = self.spectral_analysis(signal, sampling_rate)

        idx_50 = np.argmin(np.abs(freqs - 50)); feature_amp_50 =
magnitude[idx_50]
        idx_5 = np.argmin(np.abs(freqs - 5)); feature_amp_5 = magnitude[idx_5]

        magnitude_norm = magnitude / np.sum(magnitude)
        magnitude_norm_clean = magnitude_norm[magnitude_norm > 1e-10]
        feature_centroid = np.sum(freqs[:len(magnitude_norm)] *
magnitude_norm)

        feature_entropy = entropy(magnitude_norm_clean)

        return [feature_amp_50, feature_amp_5, feature_centroid, feature_entropy]

    def train_model(self, X, y):
        self.scaler.fit(X)
        X_scaled = self.scaler.transform(X)
        self.model.fit(X_scaled, y)
        self.is_trained = True
        self.training_data['X'] = X
```

```
        self.training_data['y'] = y

    def classify(self, features):
        features_scaled = self.scaler.transform([features])
        prediction = self.model.predict(features_scaled)[0]
        try:
            confidence = self.model.predict_proba(features_scaled)[0][prediction]
        except:
            confidence = 1.0
        return self.labels[prediction], confidence

class DataFusion:
    def fuse_and_decide(self, classification_result, confidence, features):
        CONFIDENCE_THRESHOLD = 0.85
        DRONE_AMPLITUDE_THRESHOLD = 0.008

        status = 'SAFE'
        threat_type = 'none'
        jamming_type = 'none'

        drone_amp = features[0]
        IOT_amp = features[1]

        if confidence >= CONFIDENCE_THRESHOLD:
            if classification_result == 'Drone':
                if drone_amp >= DRONE_AMPLITUDE_THRESHOLD:
                    status = 'DANGER'
                    threat_type = 'drone'
                    jamming_type = 'WIDEBAND'
                else:
                    status = 'WARNING'
                    threat_type = 'drone'
                    jamming_type = 'NARROWBAND'

            elif classification_result == 'IOT':
                if IOT_amp > 0.005:
                    status = 'WARNING'
                    threat_type = 'IOT'
                    jamming_type = 'NARROWBAND'
```

```
        elif drone_amp >= DRONE_AMPLITUDE_THRESHOLD and IOT_amp
>= 0.005:
            status = 'DANGER_FUSION'
            threat_type = 'mixed'
            jamming_type = 'WIDEBAND'

        else:
            status = 'SAFE_LOW_CONFIDENCE'

        return status, threat_type, jamming_type

class Mitigation:
    def __init__(self, sampling_rate=1000, duration=1.0):
        self.sampling_rate = sampling_rate
        self.t = np.linspace(0.0, duration, int(sampling_rate * duration),
endpoint=False)
        self.N = len(self.t)

    def generate_jamming_signal(self, jamming_type, strength=1.0):
        if jamming_type == 'WIDEBAND':
            jamming_power = strength * 5
            jamming_signal = np.random.randn(self.N) * np.sqrt(jamming_power)
        elif jamming_type == 'NARROWBAND':
            target_freq = 50 if strength >= 1.5 else 5
            jamming_power = strength * 3
            jamming_signal = np.sqrt(jamming_power) * np.sin(2 * np.pi *
target_freq * self.t)
        else:
            return np.zeros(self.N)

        return jamming_signal

    def apply_jamming(self, signal, jamming_signal):
        return signal + jamming_signal

    def evaluate_jamming_effectiveness(self, original_features, mitigated_signal,
detection_module, threat_type_detected):
        mitigated_features = detection_module.extract_features(mitigated_signal)
```

```
        key_index = 0 if threat_type_detected == 'drone' else 1

        original_amp = original_features[key_index]
        mitigated_amp = mitigated_features[key_index]

        if original_amp == 0:
            reduction = 0.0
        else:
            reduction = (original_amp - mitigated_amp) / original_amp

        return mitigated_features, reduction, key_index

class C2Integration:
    def __init__(self, log_output):
        self.log_output = log_output

    def log_event(self, event):
        self.log_output.insert(tk.END, f"{event}\n")
        self.log_output.see(tk.END)
```

Додаток 3 Код REBSimulatorGUI та запуск програми

```
# --- Клас GUI ---

class REBSimulatorGUI:
    def __init__(self, master):
        self.master = master
        master.title("□□ Симулятор РЕБ: БПЛА vs IOT")
        master.geometry("500x750")

        self.sensors = Sensors()
        self.detection_classification = DetectionClassification()
        self.data_fusion = DataFusion()
        self.mitigation = Mitigation()

        self.setup_ui()
        self.c2_integration.log_event("Система ініціалізована. Натисніть
'Навчити модель'.")

    def setup_ui(self):
        style = ttk.Style()
        style.configure('TFrame', background='#f0f0f0')
        style.configure('TLabel', background='#f0f0f0', font=('Arial', 10))
        style.configure('TButton', padding=6, font=('Arial', 10, 'bold'))
        style.configure('TLabelFrame', background='#e0e0e0')

        style.configure('TButton.info.TButton', background='#4CAF50',
foreground='black')
        style.map('TButton.info.TButton', background=[('active', '#66BB6A')])

        style.configure('TButton.danger.TButton', background='#F44336',
foreground='black')
        style.map('TButton.danger.TButton', background=[('active', '#E57373')])

        style.configure('TButton.warning.TButton', background='#FF9800',
foreground='black')
        style.map('TButton.warning.TButton', background=[('active', '#FFB74D')])
```

```
        style.configure('TButton.mixed.TButton', background='#FFEB3B',
foreground='black')
        style.map('TButton.mixed.TButton', background=[('active', '#FFF176')])


        main_frame = ttk.Frame(self.master, padding="15")
        main_frame.pack(fill=tk.BOTH, expand=True)

        ttk.Label(main_frame, text="□□ Радіоелектронна боротьба",
font=("Arial", 18, "bold"), background=style.lookup('TFrame',
'background')).pack(pady=10)

        param_frame = ttk.LabelFrame(main_frame, text="□ Параметри
моделювання", padding="10")
        param_frame.pack(pady=10, fill=tk.X)

        ttk.Label(param_frame, text="SNR (Природний Шум, дБ):",
background=style.lookup('TLabelFrame', 'background')).pack(pady=(5, 0))
        self.snr_var = tk.DoubleVar(value=30)
        ttk.Scale(param_frame, from_=5, to=50, orient=tk.HORIZONTAL,
variable=self.snr_var, length=400).pack()
        ttk.Label(param_frame, textvariable=self.snr_var,
background=style.lookup('TLabelFrame', 'background')).pack()

        ttk.Label(param_frame, text="Рівень Ворожої Завади (J/S в дБ):",
background=style.lookup('TLabelFrame', 'background')).pack(pady=(5, 0))
        self.hostile_jam_var = tk.DoubleVar(value=-5.0) # Діапазон J/S
        ttk.Scale(param_frame, from_=-10, to=20, orient=tk.HORIZONTAL,
variable=self.hostile_jam_var, length=400).pack()
        ttk.Label(param_frame, textvariable=self.hostile_jam_var,
background=style.lookup('TLabelFrame', 'background')).pack()

        ttk.Label(param_frame, text="Сила РЕБ (Контрзахід, Strength
Modifier):", background=style.lookup('TLabelFrame',
'background')).pack(pady=(5, 0))
        self.mitigation_strength_var = tk.DoubleVar(value=1.5) # Перейменована
змінна
        ttk.Scale(param_frame, from_=0.5, to=3.0, orient=tk.HORIZONTAL,
variable=self.mitigation_strength_var, length=400).pack()
```

```
        ttk.Label(param_frame, textvariable=self.mitigation_strength_var,
background=style.lookup('TLabelFrame', 'background')).pack()

        control_frame = ttk.LabelFrame(main_frame, text="    Управління
системою", padding="10")
        control_frame.pack(pady=10, fill=tk.X)

        train_viz_frame = ttk.Frame(control_frame)
        train_viz_frame.pack(pady=5, fill=tk.X)

        ttk.Button(train_viz_frame, text="    Навчити модель",
                   command=self.train_initial_model, style='TButton.info.TButton',
width=20).pack(side=tk.LEFT, padx=5, expand=True)

        ttk.Button(train_viz_frame, text="    Графік навчання",
                   command=self.show_training_plot, style='TButton.info.TButton',
width=20).pack(side=tk.LEFT, padx=5, expand=True)



        sim_frame = ttk.Frame(control_frame)
        sim_frame.pack(pady=10, fill=tk.X)

        ttk.Button(sim_frame, text="    БПЛА (Drone)",
                   command=lambda: self.run_simulation('drone'), width=15,
                   style='TButton.danger.TButton').pack(side=tk.LEFT, padx=5,
expand=True)

        ttk.Button(sim_frame, text="    IOT",
                   command=lambda: self.run_simulation('IOT'), width=15,
                   style='TButton.warning.TButton').pack(side=tk.LEFT, padx=5,
expand=True)

        ttk.Button(sim_frame, text="□ Змішана загроза",
                   command=lambda: self.run_simulation('mixed'), width=15,
                   style='TButton.mixed.TButton').pack(side=tk.LEFT, padx=5,
expand=True)

        ttk.Label(main_frame, text="    Лог подій C2:", font=("Arial", 12, "bold"),
background=style.lookup('TFrame', 'background')).pack(pady=(10, 0))
```

```
        self.log_output = scrolledtext.ScrolledText(main_frame, wrap=tk.WORD,
height=18, width=50)
        self.log_output.pack(pady=10, fill=tk.BOTH, expand=True)
        self.c2_integration = C2Integration(self.log_output)


    def train_initial_model(self):
        self.c2_integration.log_event("□ Початок навчання моделі...")

        X = []
        y = []
        for snr in [10, 20, 30, 40]:
            for _ in range(50):

X.append(self.detection_classification.extract_features(self.sensors.generate_sig
nal('drone', snr, None)[1]))
                y.append(1)

X.append(self.detection_classification.extract_features(self.sensors.generate_sig
nal('IOT', snr, None)[1]))
                y.append(0)

        self.detection_classification.train_model(np.array(X), np.array(y))
        self.c2_integration.log_event("□ Модель класифікації (SVC) навчена та
готова до роботи.")

    def show_training_plot(self):
        if not self.detection_classification.is_trained:
            messagebox.showerror("Помилка", "Будь ласка, спочатку натисніть
'Навчити модель'.")
            return

        X_train = self.detection_classification.training_data['X']
        y_train = self.detection_classification.training_data['y']

        X_plot = X_train[:, [0, 1]]

        scaler_plot = StandardScaler()
        X_plot_scaled = scaler_plot.fit_transform(X_plot)
```

```
        model_plot = SVC(kernel='linear')
        model_plot.fit(X_plot_scaled, y_train)

        x_min, x_max = X_plot_scaled[:, 0].min() - 0.5, X_plot_scaled[:, 0].max()
+ 0.5
        y_min, y_max = X_plot_scaled[:, 1].min() - 0.5, X_plot_scaled[:, 1].max()
+ 0.5
        xx, yy = np.meshgrid(np.arange(x_min, x_max, 0.02),
                             np.arange(y_min, y_max, 0.02))

        Z = model_plot.predict(np.c_[xx.ravel(), yy.ravel()])
        Z = Z.reshape(xx.shape)

        plot_window = tk.Toplevel(self.master)
        plot_window.title("    Межа рішення класифікатора (SVC)")

        plt.style.use('default')
        fig, ax = plt.subplots(figsize=(8, 6))

        ax.contourf(xx, yy, Z, alpha=0.3, cmap=plt.cm.coolwarm)

        scatter = ax.scatter(X_plot_scaled[:, 0], X_plot_scaled[:, 1], c=y_train,
                             cmap=plt.cm.coolwarm, edgecolors='k')

        ax.set_xlabel("Масштабована амплітуда 50 Гц (Ключова ознака
БПЛА)")
        ax.set_ylabel("Масштабована амплітуда 5 Гц (Ключова ознака IOT)")
        ax.set_title("Розділення класів 'Drone' (Червоний) та 'IOT' (Синій)")

        handles, _ = scatter.legend_elements()
        legend1 = ax.legend(handles, ['IOT', 'Drone'],
                            title="Класи",
                            loc="upper right")
        ax.add_artist(legend1)

        canvas = FigureCanvasTkAgg(fig, master=plot_window)
        canvas_widget = canvas.get_tk_widget()
        canvas_widget.pack(side=tk.TOP, fill=tk.BOTH, expand=True)
```

```
        toolbar = NavigationToolbar2Tk(canvas, plot_window)
        toolbar.update()


    def run_simulation(self, threat_type):
        if not self.detection_classification.is_trained:
            messagebox.showerror("Помилка", "Будь ласка, спочатку натисніть
'Навчити модель'.")
            return

        self.log_output.delete('1.0', tk.END)
        self.c2_integration.log_event(f"--- СТАРТ СИМУЛЯЦІЇ
'{threat_type.upper()}' ---")

        snr_db = self.snr_var.get()
        hostile_jam_db = self.hostile_jam_var.get() # Рівень ворожої завади
        mitigation_strength = self.mitigation_strength_var.get() # Сила РЕБ-
контрзаходу

        t, original_signal, sinr_db = self.sensors.generate_signal(
            threat_type, snr_db, hostile_js_db=hostile_jam_db
        )
        original_features =
self.detection_classification.extract_features(original_signal)
        freqs, magnitude_orig =
self.detection_classification.spectral_analysis(original_signal)

        self.c2_integration.log_event(f"    Якість прийому (SINR): {sinr_db:.2f}
дБ")

        classification_result, confidence =
self.detection_classification.classify(original_features)

        decision, threat_type_detected, jamming_type =
self.data_fusion.fuse_and_decide(
            classification_result, confidence, original_features
        )
```

```
        is_jamming_applied = (decision == 'DANGER' or decision ==
'WARNING' or decision == 'DANGER_FUSION')

        self.c2_integration.log_event(f"    Класифікація: {classification_result}
(Впевненість: {confidence*100:.1f}%)")
        self.c2_integration.log_event(f"□ Рішення С2: {decision} (Тип РЕБ:
{jamming_type if is_jamming_applied else 'N/A'})")

        if is_jamming_applied:
            jamming_signal =
self.mitigation.generate_jamming_signal(jamming_type,
strength=mitigation_strength)
            mitigated_signal = self.mitigation.apply_jamming(original_signal,
jamming_signal)

            mitigated_features, reduction_rate, key_index =
self.mitigation.evaluate_jamming_effectiveness(
                original_features, mitigated_signal, self.detection_classification,
threat_type_detected
            )
            freqs, magnitude_mitigated =
self.detection_classification.spectral_analysis(mitigated_signal)

            re_classification, re_confidence =
self.detection_classification.classify(mitigated_features)

            self.c2_integration.log_event(f"□□ Застосовано {jamming_type}-РЕБ
(Сила: {mitigation_strength:.1f})")
            self.c2_integration.log_event(f"    Зниження ключової ознаки:
{reduction_rate * 100:.2f}%")
            self.c2_integration.log_event(f"    Re-класифікація: {re_classification}
(Впевненість: {re_confidence*100:.1f}%)")

        else:
            self.c2_integration.log_event("Система працює в штатному режимі
або низька впевненість. РЕБ не застосовано.")
            jamming_signal = np.zeros_like(t)
            mitigated_signal = original_signal
            mitigated_features = original_features
```

```
            magnitude_mitigated = magnitude_orig
            key_index = 0

        signals = {
            't': t, 'orig': original_signal, 'jam': jamming_signal, 'mitigated':
mitigated_signal,
            'freqs': freqs, 'mag_orig': magnitude_orig, 'mag_mitigated':
magnitude_mitigated,
            'orig_feat': original_features, 'mit_feat': mitigated_features, 'threat':
threat_type,
            'key_index': key_index, 'jam_type': jamming_type, 'is_jamming_applied':
is_jamming_applied,
            'sinr_db': sinr_db
        }

        self.c2_integration.log_event("--- СИМУЛЯЦІЯ ЗАВЕРШЕНА ---")

        self.show_plots(signals, decision)

    def show_plots(self, signals, decision):
        plot_window = tk.Toplevel(self.master)
        plot_window.title(f"    Результати Симуляції: {signals['threat'].upper()}
(SINR: {signals['sinr_db']:.2f} дБ)")

        plt.style.use('seaborn-v0_8-deep')

        fig, axes = plt.subplots(3, 2, figsize=(11, 9))
        plt.tight_layout(pad=3.0)

        t, freqs = signals['t'], signals['freqs']
        mag_orig, mag_mit = signals['mag_orig'], signals['mag_mitigated']
        orig_feat, mit_feat = signals['orig_feat'], signals['mit_feat']
        key_index = signals['key_index']
        is_jamming_applied = signals['is_jamming_applied']

        axes[0, 0].plot(t, signals['orig'], color='blue')
        axes[0, 0].set_title(f"1. Сигнал {signals['threat'].upper()} (Отриманий,
SINR: {signals['sinr_db']:.2f} дБ)"); axes[0, 0].grid(True)
        axes[0, 0].set_xlabel("Час (с)"); axes[0, 0].set_ylabel("Амплітуда")
```

```
        axes[0, 1].plot(freqs, mag_orig, color='darkblue')
        axes[0, 1].set_title(f"2. Спектр {signals['threat'].upper()} (ДО РЕБ)");
axes[0, 1].set_xlim(0, 150); axes[0, 1].grid(True)
        axes[0, 1].set_xlabel("Частота (Гц)"); axes[0, 1].set_ylabel("Магнітуда
БПФ")

        axes[2, 0].plot(t, signals['mitigated'], color='purple')
        axes[2, 0].set_title("5. Сигнал ПІСЛЯ РЕБ (Остаточний)"); axes[2,
0].grid(True)
        axes[2, 0].set_xlabel("Час (с)"); axes[2, 0].set_ylabel("Амплітуда")

        if is_jamming_applied:
            axes[1, 0].plot(t, signals['jam'], color='red')
            axes[1, 0].set_title(f"3. Сигнал РЕБ-Контрзаходу
({signals['jam_type']})"); axes[1, 0].grid(True)
            axes[1, 0].set_xlabel("Час (с)"); axes[1, 0].set_ylabel("Амплітуда")

            axes[1, 1].plot(freqs, mag_orig, label='ДО РЕБ', alpha=0.8, linestyle='--
')
            axes[1, 1].plot(freqs, mag_mit, label='ПІСЛЯ РЕБ', alpha=0.8,
color='red')
            axes[1, 1].set_title("4. Порівняння спектрів (Ефект РЕБ)")
            axes[1, 1].set_xlim(0, 150); axes[1, 1].legend(); axes[1, 1].grid(True)
            axes[1, 1].set_xlabel("Частота (Гц)")

            key_label = f"Амплітуда {50 if key_index == 0 else 5} Гц"
            axes[2, 1].bar(['ДО РЕБ', 'ПІСЛЯ РЕБ'], [orig_feat[key_index],
mit_feat[key_index]], color=['skyblue', 'red'])
            axes[2, 1].set_title(f"6. Вплив РЕБ на ключову ознаку: {key_label}")
            axes[2, 1].set_ylabel("Амплітуда"); axes[2, 1].grid(axis='y', alpha=0.5)

        else:
            axes[1, 0].set_title("3. РЕБ не застосовано (SAFE)"); axes[1,
0].set_xticks([]); axes[1, 0].set_yticks([]); axes[1, 0].text(0.5, 0.5, "РЕБ не
застосовано", transform=axes[1, 0].transAxes, ha='center', va='center',
fontsize=14, color='gray')
            axes[1, 1].set_title("4. Спектр без змін"); axes[1, 1].plot(freqs,
mag_orig, color='darkgray'); axes[1, 1].set_xlim(0, 150); axes[1, 1].grid(True)
```

```
            axes[1, 1].set_xlabel("Частота (Гц)")

            axes[2, 1].set_title("6. Ознаки без змін");
            axes[2, 1].bar(['ДО РЕБ', 'ПІСЛЯ РЕБ'], [orig_feat[0], orig_feat[0]],
color=['gray', 'gray'])
            axes[2, 1].set_ylabel("Амплітуда"); axes[2, 1].grid(axis='y', alpha=0.5)


        canvas = FigureCanvasTkAgg(fig, master=plot_window)
        canvas_widget = canvas.get_tk_widget()
        canvas_widget.pack(side=tk.TOP, fill=tk.BOTH, expand=True)

        toolbar = NavigationToolbar2Tk(canvas, plot_window)
        toolbar.update()

        canvas.draw()


if __name__ == '__main__':
    root = tk.Tk()
    app = REBSimulatorGUI(root)
    root.mainloop()
```